

**RECOPIACIÓN PARA ESTUDIANTES
DE LA UNIVERSIDAD DE PANAMÁ
FAECO**



**LIC. JORGE KARICA C.
PROFESOR DE INTRODUCCIÓN A LA INFORMÁTICA
FACULTAD DE ADMINISTRACIÓN DE EMPRESAS
Y CONTABILIDAD
UNIVERSIDAD DE PANAMÁ**

QUÉ ES CADA TECNOLOGÍA

Qué es HTML

HTML es el lenguaje con el que se definen las páginas web. Básicamente se trata de un conjunto de etiquetas que sirven para definir la forma en la que presentar el texto y otros elementos de la página.

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, si 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 febrero de 2001.

El HTML es un lenguaje de programación muy fácil de aprender, lo que permite que cualquier persona, aunque no haya programado en la vida pueda enfrentarse a la tarea de crear una web. HTML es fácil y pronto podremos dominar el lenguaje. Más adelante se conseguirán los resultados profesionales gracias a nuestras capacidades para el diseño y nuestra vena artista.

Una vez conocemos el concepto de HTML os vamos a adelantar algunas cosas más. Este lenguaje se escribe en un documento de texto, por eso necesitamos un editor de textos para escribir una página web. Así pues, el archivo donde está contenido el código HTML es un archivo de texto, con una peculiaridad, que tiene extensión .html o .htm (es indiferente cuál utilizar). De modo que cuando programemos en HTML lo haremos con un editor de textos, lo más sencillo posible y guardaremos nuestros trabajos con extensión .html, por ejemplo mipagina.html

Por adelantar un poco cómo se utiliza el HTML os diremos que el lenguaje consta de etiquetas que tienen esta forma o <P>. Cada etiqueta significa una cosa, por ejemplo significa que se escriba en negrita (bold) o <P> significa un párrafo, <A> es un enlace, etc. Casi todas las etiquetas tienen su correspondiente etiqueta de cierre, que indica que a partir de ese punto no debe de afectar la etiqueta. Por ejemplo se utiliza para indicar que se deje de escribir en negrita. Así que el HTML no es más que una serie de etiquetas que se utilizan para definir la forma o estilo que queremos aplicar a nuestro documento. **Esto está en negrita**.

Para aprender [HTML en profundidad tenemos un manual en DesarrolloWeb.com](#). Además se pueden consultar los enlaces a distintos manuales que tenemos en nuestro [buscador en la sección de HTML](#).

Si lo que deseamos es tener una idea global de lo que es la publicación en Internet y los pasos a seguir para colocar nuestras páginas en la web lo más adecuado será consultar el manual de [Publicar en Internet](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es DHTML

A medida que vamos avanzando en la programación de páginas web nos vamos fijando nuevos objetivos para crear cada día webs más excitantes. Siguiendo este camino, **llega un momento que el lenguaje HTML se nos queda corto y tenemos que servirnos de alguna tecnología superior, que nos permita realizar esos desarrollos más complejos y dinámicos.**

Imaginaros por un momento que tuvieseis entre manos un gran proyecto, un proyecto que supusiese la creación masiva de páginas, como puede ser un periódico, donde cada día hay que cambiar los contenidos por completo, o una enciclopedia online, con miles de páginas y referencias, por poner dos ejemplos. Si utilizásemos únicamente HTML necesitaríamos un regimiento de maquetadores web para poder llevar a cabo el trabajo de crear tantas y tantas páginas y su actualización.

Así mismo, si quisiésemos desarrollar una aplicación en la web donde el usuario tuviese que interactuar con la página, o una aplicación que ofreciese algún servicio, como un buscador o un gestor de correo a través de la web, también nos veríamos muy limitados con el HTML.

Además, también estamos muy limitados con el HTML a la hora de crear efectos en las páginas, animaciones que llamen un poco la atención del usuario y que permitan hacer que las páginas web sean más divertidas.

DHTML es lo que hace posible crear una páginas web que salven todas las limitaciones del HTML como las comentadas con anterioridad. Como vemos, el DHTML es muy amplio y **engloba muchas técnicas que se pueden realizas con multitud de lenguajes de programación y programas distintos** .

Vamos a hacer una clasificación de DHTML para acotar un poco sus radios de acción y para que el concepto se acote en áreas de la programación web que podemos ya conocer.

DHTML de cliente

Por un lado tenemos el **DHTML que se desarrolla en el ámbito de una página web, cuando la página se está viendo en la pantalla de los usuarios** , es decir, en los navegadores. En estos casos, para realizar cualquier tipo de efecto o interactividad en la página tenemos como recurso al navegador, por eso se llama de cliente.

La programación en el cliente sirve para muchas cosas, ejemplos de ello son efectos diversos en las páginas, sonidos, videos, menús interactivos, control y respuesta a las acciones de un usuario en la página, control sobre los formularios, etc. Para hacer muchas de estas cosas podemos utilizar diversos lenguajes de programación como **Javascript** y **VBScript**, o incluso podemos meter aquí programas como **Flash**.

No obstante está más cercana a la idea del DHTML el programar scripts dentro de la página con los lenguajes del lado del cliente. **Javascript** para todos los navegadores y **VBScript** para Internet Explorer. Estos lenguajes trabajan, como se ha dicho, integrados con el navegador y dependen del modelo y de la versión de éste.

Estos lenguajes no permiten el desarrollo de cualquier proyecto en Internet, ya que al ser ejecutados en el navegador del cliente, no tienen acceso a todos los recursos del sistema del usuario, para evitar agujeros de seguridad, ni a los recursos del servidor donde están alojadas las páginas. Esta limitación, añadida a la ya comentada de su dependencia del navegador, los hacen insuficientes para desarrollos avanzados, siendo más bien un complemento de programación que el núcleo de verdaderas aplicaciones en el web.

DHTML de servidor

Por otro lado, existen una serie de **lenguajes que se basan en el servidor para ejecutar sus scripts**, al igual que la programación del cliente se basa en el navegador. **Cuando una página es solicitada por parte de un cliente, el servidor ejecuta los scripts y genera una página resultado, que envía al cliente. La página resultado contiene únicamente código HTML**, por lo que puede ser interpretada por cualquier navegador sin lugar a errores, independientemente de su versión.

Esta independencia del navegador ya es una ventaja significativa con respecto a la programación en el cliente, pero lo es aun más que contamos con todos los recursos del servidor donde están alojadas las páginas. Estos recursos, como podrían ser gestores de bases de datos, servidores de correo o el propio sistema de archivos del servidor, son los que nos van a permitir construir todo tipo de aplicaciones.

Como ventajas adicionales se puede destacar que el código de las páginas con los scripts nunca llega al cliente, recordamos que al navegador sólo le llega HTML, y esto implica que nuestros visitantes nunca van a poder acceder al corazón de las aplicaciones que hayamos desarrollado, es decir, a los scripts del lado del servidor.

Lenguajes del lado del servidor son **ASP**, desarrollado por Microsoft, **PHP** de código libre, **JSP** para programar en **Java**, o alguna otra interfaz como **CGI**, que se desarrolla en lenguajes como **C** o **Perl**.



Esquema del funcionamiento de las páginas con scripts del servidor

Para **tratar en extensión el tema del DHTML** tenemos un manual que ofrece mucha más información. Es el [manual de páginas dinámicas](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es CSS

CSS, es una tecnología que nos permite crear páginas web de una manera más exacta. Gracias a las CSS somos mucho más dueños de los resultados finales de la página, pudiendo hacer muchas cosas que no se podía hacer utilizando solamente HTML, como incluir márgenes, tipos de letra, fondos, colores...

CSS son las siglas de Cascading Style Sheets, en español Hojas de estilo en Cascada. En este reportaje vamos a ver algunos de los efectos que se pueden crear con las CSS sin necesidad de conocer la tecnología entera.

Para empezar

Las Hojas de Estilo en Cascada se escriben dentro del código HTML de la página web, solo en casos avanzados se pueden escribir en un archivo a parte y enlazar la página con ese archivo. En un principio vamos a utilizar la manera más directa de aplicar estilos a los elementos de la página, mas adelante veremos la declaración en archivos externos. Para ello, y esto es la primera lección de este artículo, vamos a conocer un nuevo atributo que se puede utilizar en casi todas las etiquetas HTML: style.

Ejemplo:

```
<p style="color:green;font-weight:bold">El párrafo saldrá con color verde y en negrita</p>
```

Dentro del atributo style se deben indicar los atributos de estilos CSS separados por punto y coma (;). Durante este artículo vamos a conocer muchos atributos de CSS, los dos primeros que hemos visto aquí son:

Color: indica el color del contenido la etiqueta donde estemos utilizándolo, generalmente indica el color del texto.

Font-weight: indica el grosor del texto. Bold sirve para poner en negrita.

Color en los enlaces

Con HTML definimos el color de los enlaces en la etiqueta <a>, con lo atributos link, vlink y alink. Esto nos permite cambiar el color de los enlaces para todo el documento, pero ¿Y si queremos cambiar el color de un enlace en concreto, para que tenga otro color que el definido en la etiqueta <a>?

Para hacer esto utilizaremos el atributo style dentro del enlace:

```
<a href="mienlace.html" style="color:red">
```

Así saldrá el enlace en color rojo, independientemente de lo definido para todo el documento.

Espaciado entre líneas

Con CSS podemos definir el espacio que hay entre cada línea del documento, utilizando el atributo `line-height`. Por ejemplo, podemos definir que para todo un párrafo el espacio entre cada una de sus líneas sea 25 pixels:

```
<p style="line-height: 25px;">  
Un párrafo normal en el que cada una de las líneas está separada 25 pixels de la otra. Hay que poner  
suficiente texto como para que se vean 2 líneas, así saldrán separadas  
</p>
```

Espaciado entre caracteres

Se puede definir también el espacio entre cada carácter. Esto se hace con el atributo de CSS `letter-spacing`. Veamos un ejemplo:

```
<p style="letter-spacing: 12cm">  
Este párrafo tiene las letras espaciadas por 1 centímetro.  
</p>
```

Este atributo, al igual que ocurre con muchos otros de CSS, no está soportado por todos los navegadores. En concreto Netscape, en su versión 4 todavía no lo incluye.

Enlaces sin subrayado

Uno de los efectos más significativos y fáciles de realizar con CSS es eliminar el subrayado de los enlaces de una página web. Existe un atributo que sirve para definir la decoración de un texto, si está subrayado, tachado, o si no tiene ninguna de estas "decoraciones". Es el atributo `text-decoration`, en este caso indicaremos en un enlace que no queremos decoración:

```
<a href="mipagina.html" style="text-decoration:none">
```

Incluir estilos para todo un sitio web

Una de las características más potentes de la programación con hojas de estilo consiste en definir los estilos de todo un sitio web. Esto se consigue creando un archivo donde tan sólo colocamos las declaraciones de estilos de la página y enlazando todas las páginas del sitio con ese archivo. De este modo, todas las páginas comparten una misma declaración de estilos y, por tanto, si la cambiamos, cambiarán todas las páginas.

Veamos ahora todo el proceso para incluir estilos con un fichero externo.

1- Creamos el fichero con la declaración de estilos

Es un fichero de texto normal, que puede tener cualquier extensión, aunque le podemos asignar la extensión `.css` para aclararnos qué tipo de archivo es. El texto que debemos incluir debe ser escrito exclusivamente en sintaxis CSS, es un poco distinta que la sintaxis que utilizamos dentro del atributo `style`. Sería erróneo incluir

código HTML en este archivo: etiquetas y demás. Podemos ver un ejemplo a continuación.

```
P {
font-size : 12pt;
font-family : arial,Helvetica;
font-weight : normal;
}
H1 {
font-size : 36pt;
font-family : verdana,arial;
text-decoration : underline;
text-align : center;
background-color : Teal;
}
BODY {
background-color : #006600;
font-family : arial;
color : White;
}
```

2- Enlazamos la página web con la hoja de estilos

Para ello vamos a colocar la etiqueta <LINK> con los atributos

- rel="STYLESHEET" indicando que el enlace es con una hoja de estilo.
- type="text/css" porque el archivo es de texto, en sintaxis CSS.
- href="estilos.css" indica el nombre del fichero fuente de los estilos.

Veamos una página web entera que enlaza con la declaración de estilos anterior:

```
<html>
<head>
<link rel="STYLESHEET" type="text/css" href="estilos.css">
<title>Página que lee estilos</title>
</head>
<body>
<h1>Página que lee estilos</h1>
<p>
Esta página tiene en la cabecera la etiqueta necesaria para enlazar con la hoja de estilos. Es muy fácil.
</p>
</body>
</html>
```

Las CSS tienen mucho más juego

Las Hojas de Estilo en Cascada son un estándar muy amplio, con unas especificaciones y posibilidades muy grandes. En este artículo hemos visto unos cuantos efectos interesantes que realizar aunque no tengamos ningún conocimiento previo. Sin embargo, lo mejor para trabajar con esta tecnología es conocerla bien, gracias a ello, los resultados serán mucho más sorprendentes.

Para ampliar esta información y conocer más sobre CSS se puede encontrar un manual en desarrolloweb.com www.desarrolloweb.com/manuales/2

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es Javascript

Javascript es un lenguaje de programación utilizado para crear pequeños programitas encargados de realizar acciones dentro del ámbito de una página web.

Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado.

Con Javascript podemos crear efectos especiales en las páginas y definir interactividades con el usuario. El navegador del cliente es el encargado de interpretar las instrucciones Javascript y ejecutarlas para realizar estos efectos e interactividades, de modo que el mayor recurso, y tal vez el único, con que cuenta este lenguaje es el propio navegador.

Javascript es el siguiente paso, después del HTML, que puede dar un programador de la web que decida mejorar sus páginas y la potencia de sus proyectos. Es un lenguaje de programación bastante **sencillo y pensado para hacer las cosas con rapidez**, a veces con ligereza. Incluso las personas que no tengan una experiencia previa en la programación podrán aprender este lenguaje con facilidad y utilizarlo en toda su potencia con sólo un poco de práctica.

Entre las acciones típicas que se pueden realizar en Javascript tenemos dos vertientes. Por un lado los **efectos especiales** sobre páginas web, para crear contenidos dinámicos y elementos de la página que tengan movimiento, cambien de color o cualquier otro dinamismo. Por el otro, javascript nos permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que podemos crear **páginas interactivas** con programas como calculadoras, agendas, o tablas de cálculo.

Javascript es un lenguaje con muchas posibilidades, permite la programación de pequeños scripts, pero también de programas más grandes, orientados a objetos, con funciones, estructuras de datos complejas, etc. Además, Javascript pone a disposición del programador todos los elementos que forman la página web, para que éste pueda acceder a ellos y modificarlos dinámicamente.

Con Javascript el programador, que se convierte en el verdadero dueño y controlador de cada cosa que ocurre en la página cuando la está visualizando el cliente.

Ver también: Para que quede claro el lenguaje y alguna aplicación práctica, que se puede hacer y entender rápidamente, se puede acceder al artículo [Efectos Rápidos con Javascript](#). En dicho artículo veremos la implementación de un botón de volver y la muestra de la última modificación de una página web.

Referencias

En DesarrolloWeb.com hemos publicado un [manual de programación en Javascript](#), donde explicamos toda la sintaxis y metodología de programación.

Además, podemos acceder al [Manual de Javascript II](#), donde vamos a tratar de acercarnos a este lenguaje en profundidad y conocer todos sus secretos y recursos disponibles.

También hemos hecho una recopilación de scripts interesantes para hacer directamente una variada gama de efectos y utilidades para páginas web. En nuestro [Taller de Javascript](#).

Además de estos manuales, tenemos muchos otros recursos interesantes sobre el lenguaje en nuestro [buscador, en la sección Javascript](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es Visual Basic Script

Es un lenguaje de programación de scripts del lado del cliente, pero sólo compatible con Internet Explorer. Es por ello que su utilización está desaconsejada a favor de Javascript.

Está basado en Visual Basic, un popular lenguaje para crear aplicaciones Windows. Tanto su sintaxis como la manera de trabajar están muy inspirados en él. Sin embargo, no todo lo que se puede hacer en Visual Basic lo podremos hacer en Visual Basic Script, pues este último es una versión reducida del primero.

El modo de funcionamiento de Visual Basic Script para construir efectos especiales en páginas web es muy similar al utilizado en Javascript y los recursos a los que se puede acceder también son los mismos: el navegador.

Como decimos, no debemos utilizar este lenguaje en la mayoría de las ocasiones, aunque un caso donde tendría sentido utilizar Visual Basic Script sería la construcción de una Intranet donde sepamos con toda seguridad que los navegadores que se van a conectar serán siempre Internet Explorer. En este caso, un programador habitual de Visual Basic tendría más facilidades para realizar los scripts utilizando Visual Basic Script en lugar de Javascript.

Nota: El popular [ASP \(Active Server Pages\)](#) es una tecnología de programación del lado del servidor. Habitualmente, los scripts ASP se escriben con Visual Basic Script también y eso no nos debe liar. Visual Basic Script, por tanto, es un lenguaje que se puede utilizar para la programación en el cliente, pero también para la programación en el servidor.

En este artículo hemos hablado del lenguaje en su faceta del lado del cliente, puesto que en la faceta del servidor tenemos muchos manuales, pero están englobados dentro de la [programación en ASP](#).

Referencias

En DesarrolloWeb.com tenemos un [manual de Visual Basic Script](#), que resulta ser uno de los pocos que se encuentran en la Red sobre esta materia. Está orientado a enseñar la sintaxis y la programación del lado del cliente con el lenguaje.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es Java

Java es un lenguaje de programación con el que podemos realizar cualquier tipo de programa. En la actualidad es un lenguaje muy extendido y cada vez cobra más importancia tanto en el ámbito de Internet como en la informática en general. Está desarrollado por la compañía Sun Microsystems con gran dedicación y siempre enfocado a cubrir las necesidades tecnológicas más punteras.

Una de las principales características por las que Java se ha hecho muy famoso es que es un lenguaje independiente de la plataforma. Eso quiere decir que si hacemos un programa en Java podrá funcionar en cualquier ordenador del mercado. Es una ventaja significativa para los desarrolladores de software, pues antes tenían que hacer un programa para cada sistema operativo, por ejemplo Windows, Linux, Apple, etc. Esto lo consigue porque se ha creado una Máquina de Java para cada sistema que hace de puente entre el sistema operativo y el programa de Java y posibilita que este último se entienda perfectamente.

La independencia de plataforma es una de las razones por las que Java es interesante para Internet, ya que muchas personas deben tener acceso con ordenadores distintos. Pero no se queda ahí, Java está desarrollándose incluso para distintos tipos de dispositivos además del ordenador como móviles, agendas y en general para cualquier cosa que se le ocurra a la industria.

Pasado y presente

Java fue pensado originalmente para utilizarse en cualquier tipo de electrodoméstico pero la idea fracasó. Uno de los fundadores de Sun rescató la idea para utilizarla en el ámbito de Internet y convirtieron a Java en un lenguaje potente, seguro y universal gracias a que lo puede utilizar todo el mundo y es gratuito. Una de los primeros triunfos de Java fue que se integró en el navegador Netscape y permitía ejecutar programas dentro de una página web, hasta entonces impensable con el HTML.

Actualmente Java se utiliza en un amplio abanico de posibilidades y casi cualquier cosa que se puede hacer en cualquier lenguaje se puede hacer también en Java y muchas veces con grandes ventajas. Para lo que nos interesa a nosotros, con Java podemos programar páginas web dinámicas, con accesos a bases de datos, utilizando XML, con cualquier tipo de conexión de red entre cualquier sistema. En general, cualquier aplicación que deseemos hacer con acceso a través web se puede hacer utilizando Java.

Conclusión

En nuestro [buscador en la sección de Java](#) podemos encontrar muchas páginas con algunos tutoriales y fuentes de referencia muy útiles. Java es tan grande que si queremos hacer cosas complicadas no podremos aprenderlo todo a través de los tutoriales online y nos harán falta libros o cursos. La página de [Java de Sun](#) es sin duda la mejor referencia, aunque un poco avanzada para los que empiezan, es la referencia más útil para los que necesitan documentarse para algo en concreto.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué son los Applets de Java

Es otra manera de incluir código a ejecutar en los clientes que visualizan una página web. Se trata de pequeños programas hechos en Java, que se transfieren con las páginas web y que el navegador ejecuta en el espacio de la página.

Los applets de Java están programados en Java y precompilados, es por ello que la manera de trabajar de éstos varía un poco con respecto a los lenguajes de script como Javascript. Los applets son más difíciles de programar que los scripts en Javascript y requerirán unos conocimientos básicos o medios del lenguaje Java.

La principal ventaja de utilizar applets consiste en que son mucho menos dependientes del navegador que los scripts en Javascript, incluso independientes del sistema operativo del ordenador donde se ejecutan. Además, Java es más potente que Javascript, por lo que el número de aplicaciones de los applets podrá ser mayor.

Como desventajas en relación con Javascript cabe señalar que los applets son más lentos de procesar y que tienen espacio muy delimitado en la página donde se ejecutan, es decir, no se mezclan con todos los componentes de la página ni tienen acceso a ellos. Es por ello que con los applets de Java no podremos hacer directamente cosas como abrir ventanas secundarias, controlar Frames, formularios, capas, etc.

Cómo es posible la multiplataforma en Java

Java es compatible con todos los sistemas porque basa su funcionamiento en los Byte Codes, que no es más que una precompilación del código fuente de Java.

Estos Byte Codes no son el programa en Java propiamente dicho, sino un archivo que contiene un código intermedio que puede manejar la Máquina Virtual de Java. Cada sistema operativo dispone de una Máquina Virtual de Java que puede interpretar los Byte Codes y transformarlos a sentencias ejecutables en el sistema en cuestión.

Ejemplos de Applets de Java

En la web podemos encontrar muchos ejemplos de applets de Java muy útiles y complejos. Nosotros aquí vamos a mostrar un par de ejemplos que no tienen demasiada importancia, pero que pueden aclarar un poco más de qué se tratan estos applets.

- **[Ejemplo de Reloj 3D](#)**. Efecto visual para incorporar un reloj digital en 3D en tu página web.
- **[Ejemplo de banner dinámico](#)**. Una animación de diversas palabras que salen en el espacio de un banner. Se puede configurar para que salga el texto que deseemos.

Nota: Los applets no los hemos hecho nosotros.

Estos applets los hemos descargado de Internet para mostrarlos en este artículo. Si lo deseáis, podéis acceder a los archivos .zip con los applets y su código para mostrarlos en una página HTML.

- + [Reloj 3D - 5.10 Kb](#)
- + [Banner animado - 14.6 Kb](#)

Ampliación

Si deseas ampliar la información sobre los applets o sobre Java recomendamos estudiar la [sección Java en el buscador de DesarrolloWeb.com](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es ASP

ASP (Active Server Pages) es la tecnología desarrollada por Microsoft para la creación de páginas dinámicas del servidor. ASP se escribe en la misma página web, utilizando el lenguaje [Visual Basic Script](#) o Jscript (Javascript de Microsoft).

Un lenguaje del lado del servidor es aquel que **se ejecuta en el servidor web**, justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que verá el cliente. El cliente solamente recibe una página con el código HTML resultante de la ejecución de la PHP. Como la página resultante contiene únicamente código HTML, es compatible con todos los navegadores. Podemos saber algo más sobre la programación del servidor y del cliente en el artículo [qué es DHTML](#).

El tipo de servidores que emplean este lenguaje son, evidentemente, todos aquellos que funcionan con sistema Windows NT, aunque también se puede utilizar en un PC con windows 98 si instalamos un servidor denominado [Personal Web Server](#). Incluso en sistemas Linux podemos utilizar las ASP si instalamos un componente denominado [Chilisoft](#), aunque parece claro que será mejor trabajar sobre el servidor web para el que está pensado: [Internet Information Server](#).

Con las ASP podemos realizar muchos tipos de aplicaciones distintas. Nos permite acceso a bases de datos, al sistema de archivos del servidor y en general a todos los recursos que tenga el propio servidor. También tenemos la posibilidad de comprar componentes ActiveX fabricados por distintas empresas de desarrollo de software que sirven para realizar múltiples usos, como el envío de correo, generar gráficas dinámicamente, y un largo etc.

Actualmente se ha presentado ya la segunda versión de ASP, el ASP.NET, que comprende algunas mejoras en cuanto a posibilidades del lenguaje y rapidez con la que funciona. ASP.NET tiene algunas diferencias en cuanto a sintaxis con el ASP, de modo que se ha de tratar de distinta manera uno de otro.

Para enterarnos bien de lo que es ASP y aprender a programar en el lenguaje disponemos de una [sección en DesarrolloWeb dedicada por entero a tratar ASP a fondo](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es PHP

PHP es el acrónimo de Hipertext Preprocesor. **Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma**, rápido, con una gran librería de funciones y mucha documentación.

Un lenguaje del lado del servidor es aquel que **se ejecuta en el servidor web**, justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que verá el cliente. El cliente solamente recibe una página con el código HTML resultante de la ejecución de la PHP. Como la página resultante contiene únicamente código HTML, es compatible con todos los navegadores. Podemos saber algo más sobre la programación del servidor y del cliente en el artículo [qué es DHTML](#).



Esquema del funcionamiento de las páginas PHP.

Una vez que ya conocemos el concepto de lenguaje de programación de scripts del lado del servidor podemos hablar de PHP. **PHP se escribe dentro del código HTML**, lo que lo hace realmente fácil de utilizar, al igual que ocurre con el popular [ASP](#) de Microsoft, pero con algunas ventajas como su gratuidad, independencia de plataforma, rapidez y seguridad. Cualquiera puede descargar a través de la página principal de PHP www.php.net y de manera gratuita, un módulo que hace que nuestro servidor web comprenda los scripts realizados en este lenguaje. Es independiente de plataforma, puesto que existe un módulo de PHP para casi cualquier servidor web. Esto hace que cualquier sistema pueda ser compatible con el lenguaje y significa una ventaja importante, ya que permite portar el sitio desarrollado en PHP de un sistema a otro sin prácticamente ningún trabajo.

PHP, en el caso de estar montado sobre un servidor Linux u Unix, es más rápido que [ASP](#), dado que se ejecuta en un único espacio de memoria y esto evita las comunicaciones entre componentes COM que se realizan entre todas las tecnologías implicadas en una página [ASP](#).

Por último señalábamos la seguridad, en este punto también es importante el hecho de que en muchas ocasiones PHP se encuentra instalado sobre servidores Unix o Linux, que son de sobra conocidos como más veloces y seguros que el sistema operativo donde se ejecuta las [ASP](#), Windows NT o 2000. Además, PHP permite configurar el servidor de modo que se permita o rechacen diferentes usos, lo que puede hacer al lenguaje más o menos seguro dependiendo de las necesidades de cada cual.

Fue creado originalmente en 1994 por Rasmus Lerdorf, pero como **PHP está desarrollado en política de código abierto**, a lo largo de su historia ha tenido muchas contribuciones de otros desarrolladores. Actualmente PHP se encuentra en su **versión 4, que utiliza el motor Zend**, desarrollado con mayor meditación para cubrir las necesidades de las aplicaciones web actuales.

Este lenguaje de programación está preparado para realizar muchos tipos de aplicaciones web gracias a la extensa librería de funciones con la que está dotado. La librería de funciones cubre desde cálculos matemáticos complejos hasta tratamiento de conexiones de red, por poner dos ejemplos.

Algunas de las más importantes capacidades de PHP son: **compatibilidad con las bases de datos** más comunes, como [MySQL](#), [mSQL](#), [Oracle](#), [Informix](#), y [ODBC](#), por ejemplo. Incluye funciones para el **envío de correo electrónico**, **upload de archivos**, crear **dinámicamente en el servidor imágenes en formato GIF**, incluso animadas y una lista interminable de utilidades adicionales.

Se puede obtener mucha más información de PHP en los diversos manuales de PHP que hemos publicado en [desarrolloweb](#). Manuales que atienden a los diversos niveles de cada programador:

- [Programación en PHP](#): Aprende PHP desde el principio con este manual que toca las prestaciones más imprescindibles.
- [Programación en PHP II](#): Aprende a hacer una aplicación práctica en PHP. Un sistema de postales.
- [Taller de PHP](#): otros artículos prácticos más avanzados con el lenguaje.

Informe de **Miguel Angel Alvarez**
Director [desarrolloweb.com](#)
Mail: eugim@desarrolloweb.com

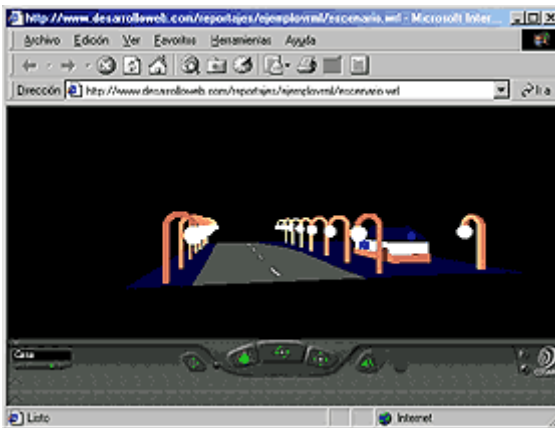
Qué es XML

XML es una tecnología en realidad muy sencilla que tiene a su alrededor otras

tecnologías que la complementan y la hacen mucho más grande y con unas posibilidades mucho mayores. Vamos a [ver a lo largo de varios capítulos una introducción al mundo XML](#), es decir, al lenguaje así como a las tecnologías que trabajan con él, sus usos, ventajas y modos de llevar a cabo las tareas.

XML, con todas las tecnologías relacionadas, representa una manera distinta de hacer las cosas, más avanzada, cuya principal novedad consiste en permitir compartir los datos con los que se trabaja a todos los niveles, por todas las aplicaciones y soportes. Así pues, el XML juega un papel importantísimo en este mundo actual, que tiende a la globalización y la compatibilidad entre los sistemas, ya que es la tecnología que permitirá compartir la información de una manera segura, fiable, fácil. Además, XML permite al programador y los soportes dedicar sus esfuerzos a las tareas importantes cuando trabaja con los datos, ya que algunas tareas tediosas como la validación de estos o el recorrido de las estructuras corre a cargo del lenguaje y está especificado por el estándar, de modo que el programador no tiene que preocuparse por ello.

Vemos que XML no está sólo, sino que hay un mundo de tecnologías alrededor de él, de posibilidades, maneras más fáciles e interesantes de trabajar con los datos y, en definitiva, un avance a la hora de tratar la información, que es en realidad el objetivo de la informática en general. XML, o mejor dicho, el mundo XML no es un



lenguaje, sino varios lenguajes, no es una sintaxis, sino varias y no es una manera totalmente nueva de trabajar, sino una manera más refinada que permitirá que todas las anteriores se puedan comunicar entre si sin problemas, ya que los datos cobran sentido. Todo esto lo veremos con calma en la [Introducción a XML](#).

XML es interesante en el mundo de Internet y el e-bussiness, ya que existen muchos sistemas distintos que tienen que comunicarse entre si, pero como se ha podido imaginar, interesa por igual a todas las ramas de la informática y el tratamiento de datos, ya que permite muchos avances a la hora de trabajar con ellos.

En la [introducción a XML](#), a lo largo de los siguientes capítulos, vamos a ver algunas características importantes de la tecnología que nos permitirán comprender mejor el mundo XML y cómo soluciona nuestros problemas a la hora de trabajar con los datos.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es VRML

El Virtual Reality Modeling Language es un **lenguaje de modelado de mundos virtuales en tres dimensiones**. Igual que el HTML nos sirve para maquetar páginas web, VRML sirve para crear mundos en tres dimensiones a los que accedemos utilizando nuestro navegador, igual que si visitásemos una página web cualquiera, con la salvedad que nuestras visitas no se limitan a ver un simple texto y fotografías, sino que nos permite ver todo tipo de objetos y construcciones en 3D por los que podemos pasear o interactuar.

Este modo de visitar sitios en Internet es mucho más avanzado y posee grandes ventajas. Para empezar **la navegación se desarrolla de una manera mucho más intuitiva**, dado que la forma de actuar dentro del mundo virtual es similar a la de la vida real. Podemos movernos en todas las direcciones, no solo izquierda y derecha sino también adelante, atrás, arriba y abajo. Podemos tratar con los objetos como en la vida misma, tocarlos, arrastrarlos, etc. y en general todo lo que podáis imaginar. También **los escenarios son mucho más reales**, pensemos en un ejemplo como podría ser una biblioteca virtual. En ella podríamos andar por cada una de las salas, tomar determinados libros y leerlos.

A la larga, el acceso a Internet se ha de convertir en una experiencia mucho más cercana a la que realizamos en la vida y las visitas a los lugares retratados en la Red serán mucho más reales. Sin embargo, en la actualidad VRML presenta muchas limitaciones con respecto a sus potencialidades, que se irán cubriendo con la llegada de máquinas más potentes y periféricos avanzados para la realidad virtual como pueden ser guantes o cascos.

Un poco de historia

El VRML surgió en la primavera de 1994, durante una reunión convocada por Tim Berners-Lee y Dave Ragget para tratar de acercar los desarrollos de realidad virtual a Internet. En esta reunión los asistentes llegaron a la conclusión que se tenía que desarrollar un lenguaje común para la descripción de los mundos en 3D.

De este modo, **en la Primera Conferencia Mundial de la WWW en Ginebra se aprobó el desarrollo de un nuevo lenguaje que permitiese crear mundos en tres dimensiones** a los que se pudiera acceder por la World Wide Web.

Con el tiempo se desarrolló el lenguaje dentro de varios requisitos: que fuese adaptable a la red, que no requiriese una línea de alta velocidad (anchos de banda elevados), que fuese multiplataforma, etc.

Materiales necesarios

Los materiales necesarios para comenzar con VRML son pocos, y posiblemente ya tengamos, sin saberlo, todos los ingredientes para introducirnos en el desarrollo y edición de mundos virtuales. Estos son:

- **Un editor de textos sencillo.** El [Block de notas](#) es ideal. También valdrá cualquier otro editor en modo ASCII. También podemos utilizar editores especializados como el [VRML PAD](#).
- **Un visualizador VRML** para ver los resultados, que se instala en el navegador como cualquier otro plug-in. Posiblemente tu navegador ya esté preparado para ver los mundos en VRML, si no es así, tienes que instalarlo. Un visualizador muy conocido es el [Cosmo Player](#).

Ejemplo de VRML

El archivo que contiene el código VRML es un fichero de texto. Este deberá ser

guardado con extensión .wrl para ser reconocido por el navegador como archivo fuente de un mundo virtual.

Para su posterior visualización simplemente habrá de abrirse con el navegador. Si nuestro visualizador se encuentra correctamente instalado se encargará de mostrar el mundo y podremos interaccionar con el.

Podemos [visitar este enlace](#) para ver un mundo en VRML, si es que tenemos instalado el visor. Si no es así no podremos ver el mundo virtual hasta que no lo hayamos instalado.

[Ver un mundo virtual](#) que construido para la ocasión.

Enlaces interesantes

1. [Directorio VRML de nuestro buscador](#)
2. [Descripción del Cosmo Player y enlace para la descarga](#)
3. [Descripción y descarga de un editor de VRML](#)

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué son las cookies

En nuestros viajes por la Red visitamos gran cantidad de páginas, muchas de ellas bastante complicadas que implementan distintos servicios de Internet. Estas páginas tienen que guardar distintas informaciones acerca de un usuario, por ejemplo su nombre, su edad o su color preferido. Para ello cuentan con una serie de mecanismos en el servidor como bases de datos u otro tipo de contenedores, pero hay un mecanismo mucho más interesante de guardar esa información que los propios recursos del servidor, que es el propio ordenador del usuario.

En nuestros ordenadores se guardan muchos datos que necesitan conocer las páginas web cada vez que entramos en la página, estas pequeñas informaciones son las cookies: **estados de variables que se conservan de una visita a otra en el ordenador del cliente.**

Como es un poco peligroso que las páginas web a las que accedemos se dediquen a introducir cosas en nuestro ordenador, las cookies están altamente restringidas. Para empezar, **solamente podemos guardar en ellas textos**, nunca programas, imágenes, etc. Además, los textos **nunca podrán ocupar más de 1 K**, de modo que nadie podría inundarnos el ordenador a base de cookies. Estas restricciones, unidas a la necesidad de **poner una fecha de caducidad** a las cookies para que estas se guarden, hacen que el aceptar cookies no signifique un verdadero problema para la integridad de nuestros sistemas.

Ejemplos de uso de las cookies

Un ejemplo típico de las cookies podría ser un **contador de las veces que accede un usuario a una página**. Podríamos poner una cookie en el ordenador del cliente donde tendríamos una variable que lleva la cuenta de las veces que ha accedido a la página y cada vez que se accede se incrementa en uno.

En desarrolloweb, igual que muchos otros sitios, utilizamos las cookies para **guardar la personalización de un usuario** de la página. En el momento de escribir este reportaje se puede ver en el diseño de desarrolloweb un botón que pone personalización y donde hay varios colores. Pulsando cada color la página cambia, para verse con ese color. Para que el color elegido se conserve de visita en visita se utilizan las cookies.

Otro ejemplo típico es el que **guarda el perfil del usuario**. Si el usuario accede a contenidos determinados podemos enviarle una cookie que le marca como interesado en un tema. A medida que va accediendo a distintos sitios le vamos encasillando como joven, adulto, hombre, mujer, o lo que proceda. Conociendo el perfil de un usuario se le pueden ofrecer un tipo de productos o servicios orientados a sus gustos o necesidades.

Cómo se utilizan las cookies

Para trabajar con cookies tenemos que **utilizar un lenguaje de programación avanzado como Javascript o ASP, PHP, etc.** No podemos entonces trabajar con cookies si solamente nos dedicamos a utilizar el HTML, que ya sabemos que es un poco limitado para cosas que se salgan de mostrar contenido en páginas estáticas.

En desarrolloweb tenemos las secciones de [ASP](#) y [PHP](#), donde puedes encontrar algunos artículos que explican el uso de las cookies en estos lenguajes.

Polémica de las cookies

Existe un problema con estas galletitas y es que nos cortan nuestra intimidad. Lo que señalábamos anteriormente acerca de guardar el perfil de un usuario puede llegar a ser un problema para nosotros, por que nos están vigilando y apuntando cada uno de nuestros movimientos, lo que puede convertirse en un abuso de información que no tiene por qué pertenecer a nadie más que nosotros. Las empresas que más utilizan esta clasificación del personal son los AD-Servers (servidores de banners) como el de Doubleclick. No queremos entrar aquí más en esta polémica, pero si despertar la inquietud de que posiblemente estén entrando en nuestra intimidad.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es SQL

Las aplicaciones en red son cada día más numerosas y versátiles. En muchos casos,

el esquema básico de operación es una serie de scripts que rigen el comportamiento de una base de datos.

Debido a la diversidad de lenguajes y de bases de datos existentes, la manera de comunicar entre unos y otros sería realmente complicada a gestionar de no ser por la existencia de estándares que nos permiten el realizar las operaciones básicas de una forma universal.

Es de eso de lo que trata el Structured Query Language que no es más que un lenguaje estándar de comunicación con bases de datos. Hablamos por tanto de un lenguaje normalizado que nos permite trabajar con cualquier tipo de lenguaje (ASP o PHP) en combinación con cualquier tipo de base de datos (MS Access, SQL Server, MySQL...).

El hecho de que sea estándar no quiere decir que sea idéntico para cada base de datos. En efecto, determinadas bases de datos implementan funciones específicas que no tienen necesariamente que funcionar en otras.

Aparte de esta universalidad, el SQL posee otras dos características muy apreciadas. Por una parte, presenta una potencia y versatilidad notables que contrasta, por otra, con su accesibilidad de aprendizaje.

[El manual de SQL de desarrolloweb](#) pretende dar a conocer las operaciones básicas que se pueden realizar con SQL y que tienen una aplicación directa con la creación de aplicaciones en red sin profundizar más de lo estrictamente necesario. Buscamos con ello ofrecer al webmaster un manual de referencia práctico y aplicado.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Qué es un Webmaster

¿Qué es Ser WebMaster? La palabra WebMaster en definitiva es una palabra de origen Inglés, que si lo traducimos al castellano queda algo como Maestro Web.

Pero muchas veces escuchamos estas palabras que soy WebMaster, conozco a un WebMaster y no sabemos realmente cual es la función específica de estas personas en las cuales me incluyo.

Un WebMaster es la persona encargada de un sitio, vendría siendo como el director de una empresa, es la persona que decide las tecnologías que se van a usar, decide, los servidores, los chiches y la estructura.

Por empezar un WebMaster es una persona responsable, es la responsable propiamente dicho de Todo un Sitio, ya que es la que tiene por ejemplo, los códigos o passwords para hacer modificaciones en la pagina. Es la única persona autorizada a que si no le gusta la estructura o los contenidos del sitio puede decir que va o no va.

WebMasters Vs. Diseñadores Gráficos.

No tenemos que confundir los tantos, los WebMaster son WebMaster propiamente dicho, un diseñador es un diseñador por lo cual, en nuestro site si no sabemos de diseño démosle trabajo a la gente que sabe del tema, ya que un WebMaster abarca otros temas como explicábamos y tiene otras responsabilidades.

Muchas veces nosotros los WebMaster queremos dar nuestro toque de diseño a nuestro site, claro que podemos aportar muchas cosas para los encargados de diseño, pero una recomendación muy importante, no decidir en el tema de diseño, tenemos que tener para armar nuestro site, un encargado en el área, un experto en el tema de diseño para que nosotros solo nos aboquemos a nuestro trabajo que es el de armar todo lo que el grupo arma, en diseño y contenido.

Una cosa esencial que tenemos que ver a la hora de empezar a trabajar en nuestro site con los diseñadores es de explicarles los servidores en los que se va a trabajar, explicarles con porcentajes las tecnologías que convienen y las que no, en definitiva adaptarlos a lo que tenemos nosotros.

WebMasters y seguridad

Si bien hablamos de que los WebMaster son las personas que están encargadas de tener los códigos de acceso a la pagina, tenemos que hablar de que también puede ser el administrador del site, por lo tanto tiene que tener conocimientos amplios sobre seguridad informática para proteger de cualquier ataque a el sitio.

También tenemos que tratar los WebMaster es de toda la información que corra por nuestro site que sean lo mas privadas posibles, tenemos que asegurarnos siempre de que nuestros Server sean seguros para que no suframos ataques de hackers con malas intenciones y que nos perjudiquen nuestro trabajo.

Pero en fin ¿Quién se puede hacer llamar WebMaster?

Para ser WebMaster no hay que estudiar ninguna carrera universitaria ni menos tenemos que ir a algún curso para recibinos de WebMaster. Los WebMaster somos personas altamente capacitadas en el ámbito de Internet por lo cual conocemos a fondo las tecnologías del mercado, somos los que en partes técnicas sabemos bien sobre lo que estamos trabajando, somos la parte mas importante del site porque somos las personas que unen a todas las personas que trabajan, los de contenido, los diseñadores, los noteros, los directivos, para hacer alguna publicación automáticamente pasan por nosotros los WebMaster.

Un punto fundamental que tenemos que tener muy pero muy en cuenta para ser

WebMaster es la responsabilidad que tenemos, ya que si leímos lo que decía mas arriba de que somos la persona mas importante de el sitio, tenemos que estar siempre listos para cualquier desafio, y tenemos que afrontar con responsabilidad nuestro puesto de trabajo.

Informe de **Mario Matías Sebely**
WebMaster ConozcaMisiones.com
Mail: webmaster@conozcamisiones.com
URL: <http://www.conozcamisiones.com>

Qué es Streaming

La tecnología de streaming se utiliza para aligerar la descarga y ejecución de audio y vídeo en la web, ya que permite escuchar y visualizar los archivos mientras se están descargando.

Si no utilizamos streaming, para mostrar un contenido multimedia en la Red, tenemos que descargar primero el archivo entero en nuestro ordenador y más tarde ejecutarlo, para finalmente ver y oír lo que el archivo contenía. Sin embargo, el streaming permite que esta tarea se realice de una manera más rápida y que podamos ver y escuchar su contenido durante la descarga.

El streaming funciona de la siguiente manera. Primero nuestro ordenador (el cliente) conecta con el servidor y éste le empieza a mandar el fichero. El cliente comienza a recibir el fichero y construye un buffer donde empieza a guardar la información. Cuando se ha llenado el buffer con una pequeña parte del archivo, el cliente lo empieza a mostrar y a la vez continúa con la descarga. El sistema está sincronizado para que el archivo se pueda ver mientras que el archivo se descarga, de modo que cuando el archivo acaba de descargarse el fichero también ha acabado de visualizarse. Si en algún momento la conexión sufre descensos de velocidad se utiliza la información que hay en el buffer, de modo que se puede aguantar un poco ese descenso. Si la comunicación se corta demasiado tiempo, el buffer se vacía y la ejecución el archivo se cortaría también hasta que se restaurase la señal.

Programas de Streaming

En realidad, este proceso de streaming lo podemos haber visto en muchas ocasiones en nuestros ordenadores. Es lo que hacen programas como el Real Player o el Windows Media Player, programas que se instalan como plug-ins en los navegadores para recibir y mostrar contenidos multimedia por streaming.

Cuando pretendemos incluir audio o video en las páginas lo mejor pues, es utilizar la tecnología de streaming. Para ello simplemente tenemos que guardar los archivos multimedia con el formato de uno de los programas de streaming y seguir unas pequeñas normas a la hora de subirlos a Internet y colocarlos en la página. Las normas que seguir son propias de cada sistema y no las veremos aquí. Lo mejor para enterarse de cómo funcionan es visitar las correspondientes páginas web, señaladas más abajo.

Para convertir los archivos de audio y vídeo al formato de cada programa de streaming se utilizan unos programas especiales que se pueden descargar de las páginas de cada tecnología. Por ejemplo, el programa para convertir al formato que lee el Real Player se llama Real Producer.

A la hora de desarrollar el web con contenidos multimedia será necesario que nos decidamos a utilizar una tecnología de streaming en concreto y no las utilizemos todas para no obligar a nuestros usuarios a descargarse todos los plug-ins del mercado. A continuación vemos las tres posibles tecnologías de streaming del momento.

Real Media es posiblemente la más popular. También es la empresa con más experiencia en el sector y desarrolla muchos productos orientados a la distribución de archivos multimedia Su web: www.real.com

Windows Media es la apuesta de Microsoft. Ya posee una cuota de usuarios muy importante y seguramente aumentará con rapidez ya que Microsoft incluye el plug-in en la instalación típica de los sistemas operativos que está fabricando.

Quick Time es la tercera en discordia. Con menor cuota de mercado.

Servidores de Streaming

En principio no es necesario contar con un servidor especial para colocar archivos de audio o vídeo con descarga streaming en nuestras webs. Cualquier servidor normal puede mandar la información y es el cliente el que se encarga de procesarla para poder mostrarla a medida que la va recibiendo.

Sin embargo, existen servidores especiales preparados para transmitir streaming. Aunque en muchas ocasiones no es necesario utilizarlos nos pueden ofrecer importantes prestaciones como mandar un archivo de mayor o menor calidad dependiendo de la velocidad de nuestra línea.

En determinados casos, como la puesta en marcha de una radio o la transmisión de un evento en directo, si que será imprescindible contar con un servidor de streaming al que mandaremos la señal y con ella, la enviará a todos los clientes a medida que la va recibiendo.

Conclusión

No cabe duda que la transmisión de contenido multimedia a través de la web será cada vez más importante. La tecnología de streaming es un mercado con futuro y grandes compañías ya están luchando por el mercado. La velocidad de Internet aumentará con el tiempo y con ella aumentará la calidad de las transmisiones, para hacer posible tanto la radio como la televisión en Internet.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es la programación orientada a objetos

La programación Orientada a objetos (POO) es una forma especial de programar, más cercana a como expresaríamos las cosas en la vida real que otros tipos de programación.

Con la POO tenemos que aprender a pensar las cosas de una manera distinta, para escribir nuestros programas en términos de objetos, propiedades, métodos y otras cosas que veremos rápidamente para aclarar conceptos y dar una pequeña base que permita soltarnos un poco con este tipo de programación.

Motivación

Durante años, los programadores se han dedicado a construir aplicaciones muy parecidas que resolvían una y otra vez los mismos problemas. Para conseguir que los esfuerzos de los programadores puedan ser utilizados por otras personas se creó la POO. Que es una serie de normas de realizar las cosas de manera que otras personas puedan utilizarlas y adelantar su trabajo, de manera que consigamos que el código se pueda reutilizar.

La POO no es difícil, pero es una manera especial de pensar, a veces subjetiva de quien la programa, de manera que la forma de hacer las cosas puede ser diferente según el programador. Aunque podamos hacer los programas de formas distintas, no todas ellas son correctas, lo difícil no es programar orientado a objetos sino programar bien. Programar bien es importante porque así nos podemos aprovechar de todas las ventajas de la POO.

Cómo se piensa en objetos

Pensar en términos de objetos es muy parecido a cómo lo haríamos en la vida real. Por ejemplo vamos a pensar en un coche para tratar de modelizarlo en un esquema de POO. Diríamos que el coche es el elemento principal que tiene una serie de características, como podrían ser el color, el modelo o la marca. Además tiene una serie de funcionalidades asociadas, como pueden ser ponerse en marcha, parar o aparcar.

Pues en un esquema POO el coche sería el objeto, las propiedades serían las características como el color o el modelo y los métodos serían las funcionalidades asociadas como ponerse en marcha o parar.

Por poner otro ejemplo vamos a ver cómo modelizaríamos en un esquema POO una fracción, es decir, esa estructura matemática que tiene un numerador y un denominador que divide al numerador, por ejemplo $3/2$.

La fracción será el objeto y tendrá dos propiedades, el numerador y el denominador. Luego podría tener varios métodos como simplificarse, sumarse con otra fracción o número, restarse con otra fracción, etc.

Estos objetos se podrán utilizar en los programas, por ejemplo en un programa de matemáticas harás uso de objetos fracción y en un programa que gestione un taller de coches utilizarás objetos coche. Los programas Orientados a objetos utilizan muchos objetos para realizar las acciones que se desean realizar y ellos mismos también son objetos. Es decir, el taller de coches será un objeto que utilizará objetos coche, herramienta, mecánico, recambios, etc.

Clases en POO

Las clases son declaraciones de objetos, también se podrían definir como abstracciones de objetos. Esto quiere decir que la definición de un objeto es la clase. Cuando programamos un objeto y definimos sus características y funcionalidades en realidad lo que estamos haciendo es programar una clase. En los

ejemplos anteriores en realidad hablábamos de las clases coche o fracción porque sólo estuvimos definiendo, aunque por encima, sus formas.

Propiedades en clases

Las propiedades o atributos son las características de los objetos. Cuando definimos una propiedad normalmente especificamos su nombre y su tipo. Nos podemos hacer a la idea de que las propiedades son algo así como variables donde almacenamos datos relacionados con los objetos.

Métodos en las clases

Son las funcionalidades asociadas a los objetos. Cuando estamos programando las clases las llamamos métodos. Los métodos son como funciones que están asociadas a un objeto.

Objetos en POO

Los objetos son ejemplares de una clase cualquiera. Cuando creamos un ejemplar tenemos que especificar la clase a partir de la cual se creará. Esta acción de crear un objeto a partir de una clase se llama instanciar (que viene de una mala traducción de la palabra instace que en inglés significa ejemplar). Por ejemplo, un objeto de la clase fracción es por ejemplo 3/5. El concepto o definición de fracción sería la clase, pero cuando ya estamos hablando de una fracción en concreto 4/7, 8/1000 o cualquier otra, la llamamos objeto.

Para crear un objeto se tiene que escribir una instrucción especial que puede ser distinta dependiendo el lenguaje de programación que se emplee, pero será algo parecido a esto.

```
miCoche = new Coche()
```

Con la palabra new especificamos que se tiene que crear una instancia de la clase que sigue a continuación. Dentro de los paréntesis podríamos colocar parámetros con los que inicializar el objeto de la clase coche.

Estados en objetos

Cuando tenemos un objeto sus propiedades toman valores. Por ejemplo, cuando tenemos un coche la propiedad color tomará un valor en concreto, como por ejemplo rojo o gris metalizado. El valor concreto de una propiedad de un objeto se llama estado.

Para acceder a un estado de un objeto para ver su valor o cambiarlo se utiliza el operador punto.

```
miCoche.color = rojo
```

El objeto es miCoche, luego colocamos el operador punto y por último el nombre e la propiedad a la que deseamos acceder. En este ejemplo estamos cambiando el valor del estado de la propiedad del objeto a rojo con una simple asignación.

Mensajes en objetos

Un mensaje en un objeto es la acción de efectuar una llamada a un método. Por ejemplo, cuando le decimos a un objeto coche que se ponga en marcha estamos pasándole el mensaje "ponte en marcha".

Para mandar mensajes a los objetos utilizamos el operador punto, seguido del método que deseamos imbocar.

```
miCoche.ponerseEnMarcha()
```

En este ejemplo pasamos el mensaje `ponerseEnMarcha()`. Hay que colocar paréntesis igual que cualquier llamada a una función, dentro irían los parámetros.

Otras cosas

Hay mucho todavía que conocer de la POO ya que sólo hemos hecho referencia a las cosas más básicas. También existen mecanismos como la herencia y el polimorfismo que son unas de las posibilidades más potentes de la POO.

La herencia sirve para crear objetos que incorporen propiedades y métodos de otros objetos. Así podremos construir unos objetos a partir de otros sin tener que reescribirlo todo.

El polimorfismo sirve para que no tengamos que preocuparnos sobre lo que estamos trabajando, y abstraernos para definir un código que sea compatible con objetos de varios tipos.

Son conceptos avanzados que cuesta explicar en las líneas de ese informe. No hay que olvidar que existen libros enteros dedicados a la POO y aquí solo pretendemos dar un repaso a algunas cosas para que os suenen cuando tengáis que ponerlos delante de ellas en los lenguajes de programación que debe conocer un desarrollador del web.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Que es un firewall

Un firewall es un dispositivo que funciona como cortafuegos entre redes, permitiendo o denegando las transmisiones de una red a la otra. Un uso típico es

situarlo entre una red local y la red Internet, como dispositivo de seguridad para evitar que los intrusos puedan acceder a información confidencial.

Un firewall es simplemente un filtro que controla todas las comunicaciones que pasan de una red a la otra y en función de lo que sean permite o deniega su paso. Para permitir o denegar una comunicación el firewall examina el tipo de servicio al que corresponde, como pueden ser el web, el correo o el IRC. Dependiendo del servicio el firewall decide si lo permite o no. Además, el firewall examina si la comunicación es entrante o saliente y dependiendo de su dirección puede permitirla o no.

De este modo un firewall puede permitir desde una red local hacia Internet servicios de web, correo y ftp, pero no a IRC que puede ser innecesario para nuestro trabajo. También podemos configurar los accesos que se hagan desde Internet hacia la red local y podemos denegarlos todos o permitir algunos servicios como el de la web, (si es que poseemos un servidor web y queremos que accesible desde Internet). Dependiendo del firewall que tengamos también podremos permitir algunos accesos a la red local desde Internet si el usuario se ha autenticado como usuario de la red local.

Un firewall puede ser un dispositivo software o hardware, es decir, un aparatito que se conecta entre la red y el cable de la conexión a Internet, o bien un programa que se instala en la máquina que tiene el modem que conecta con Internet. Incluso podemos encontrar ordenadores computadores muy potentes y con softwares específicos que lo único que hacen es monitorizar las comunicaciones entre redes.

Si se quiere saber algo más sobre firewalls podemos examinar los enlaces que tenemos en el [área de seguridad](#). Es especialmente interesante este enlace: <http://www.ciudadfutura.com/mundopc/cursos/firewalls/fire1.htm>

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es CGI

Es el sistema más antiguo que existe para la programación de las [páginas dinámicas de servidor](#). Actualmente se encuentra un poco desfasado por diversas razones entre las que destaca la dificultad con la que se desarrollan los programas y la pesada carga que supone para el servidor que los ejecuta.

Los CGI se escriben habitualmente en el lenguaje Perl, sin embargo, otros lenguajes como C, C++ o Visual Basic pueden ser también empleados para construirlos.

El funcionamiento básico de un programa CGI es parecido al apuntado para el conjunto de las páginas dinámicas del servidor, con algunas particularidades.

1. Se realiza una petición http, a la que pueden acompañar datos llegados o bien por un formulario o bien a través de la URL.
2. El servidor ejecuta los programas CGI a los que se accede y trabaja con los recursos necesarios para llevar a cabo las acciones, como por ejemplo bases de datos.
3. El programa CGI va escribiendo en la salida estándar el resultado de la ejecución del CGI, que incluye etiquetas HTML, ya que lo que se escribe es una página web.

Algunas desventajas de la programación en CGI son las siguientes:

- Los resultados se escriben directamente con el CGI, así que el código del programa se mezcla con el del HTML haciendo difícil su comprensión y mantenimiento.
- Cada programa CGI que se pone en marcha lo hace en un espacio de memoria propio. Así, si tres usuarios ponen en marcha un CGI a la vez se multiplicará por tres la cantidad de recursos que ocupe ese CGI. Esto significa una grave ineficiencia.

Para completar esta información sería interesante acceder a la [sección CGI de nuestro directorio de enlaces](#), donde podemos encontrar sitios en Internet que ofrecen tutoriales sobre la tecnología y [directorios de programas CGI](#) ya creados para hacer cosas tan variadas como tiendas, foros, envío de formularios, etc.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es Perl

Es un lenguaje de programación muy utilizado para construir aplicaciones CGI para el web. Perl es un acrónimo de Practical Extracting and Reporting Language, que viene a indicar que se trata de un lenguaje de programación muy práctico para extraer información de archivos de texto y generar informes a partir del contenido de los ficheros.

Es un lenguaje libre de uso, eso quiere decir que es gratuito. Antes estaba muy asociado a la plataforma Unix, pero en la actualidad está disponible en otros sistemas operativos como Windows.

Perl es un lenguaje de programación interpretado, al igual que muchos otros lenguajes de Internet como [Javascript](#) o [ASP](#). Esto quiere decir que el código de los scripts en Perl no se compila sino que cada vez que se quiere ejecutar se lee el código y se pone en marcha interpretando lo que hay escrito. Además es extensible a partir de otros lenguajes, ya que desde Perl podremos hacer llamadas a subprogramas escritos en otros lenguajes. También desde otros lenguajes podremos ejecutar código Perl.

Perl está inspirado a partir de lenguajes como C, sh, awk y sed (algunos provenientes de los sistemas Unix), pero está enfocado a ser más práctico y fácil que estos últimos. Es por ello que un programador que haya trabajado con el lenguaje C y los otros tendrá menos problemas en entenderlo y utilizarlo rápidamente. Una diferencia fundamental de Perl con respecto a los otros lenguajes es que no limita el tamaño de los datos con los que trabaja, el límite lo pone la memoria que en ese momento se encuentre disponible.

Si queremos trabajar con Perl será necesario tener instalado el interprete del lenguaje. A partir de ese momento podemos ejecutar CGIs en nuestros servidores web. El proceso para conseguirlo puede variar de unos servidores a otros, pero se suelen colocar en un directorio especial del servidor llamado cgi-bin donde hemos colocado los correspondientes permisos CGI. Además, los archivos con el código también deberán tener permiso de ejecución.

Este informe se complementa con los enlaces que podéis encontrar en la [sección Perl](#) de nuestro buscador y ocasionalmente la [sección CGI](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es C#

C# es el nuevo lenguaje de propósito general orientado a objetos creado por Microsoft para su nueva plataforma .NET.

Microsoft.NET es el conjunto de nuevas tecnologías en las que Microsoft ha estado trabajando estos últimos años con el objetivo de mejorar tanto su sistema operativo como su modelo de componentes (COM) para obtener una plataforma con la que sea sencillo el desarrollo de software en forma de servicios web.

Los servicios web son un novedoso tipo de componentes software que se caracterizan a la hora de trabajar por su total independencia respecto a su ubicación física real, la plataforma sobre la que corre, el lenguaje de programación con el que hayan sido desarrollados o el modelo de componentes utilizado para ello.

El acceso a estos servicios se realiza en base a estándares de Internet, como son diferentes mecanismos del protocolo HTTP (GET y PUT) o el novedoso protocolo RPC conocido como SOAP (Simple Access Object Protocol), que no es más que una combinación de estándares como HTTP y XML para realizar llamadas a los miembros de estos servicios web. La idea detrás de SOAP consiste sencillamente en utilizar HTTP como medio de transporte para el envío de los mensajes de solicitud

de ejecución de los miembros de servicios web remotos (lo que permite atravesar barreras tales como firewalls) y utilizar XML como lenguaje con el que escribir los cuerpos de estos mensajes.

Pero la plataforma .NET no son sólo los servicios web, sino que también ofrece numerosos servicios a las aplicaciones que para ella se escriban, como son un recolección de basura, independencia de la plataforma, total integración entre lenguajes (por ejemplo, es posible escribir una clase en C# que derive de otra escrita en Visual Basic.NET que a su vez derive de otra escrita en Cobol)

Como se deduce del párrafo anterior, es posible programar la plataforma .NET en prácticamente cualquier lenguaje, pero Microsoft ha decidido sacar uno nuevo porque ha visto conveniente poder disponer de un lenguaje diseñado desde 0 con vistas a ser utilizado en .NET, un lenguaje que no cuente con elementos heredados de versiones anteriores e innecesarios en esta plataforma y que por tanto sea lo más sencillo posible para programarla aprovechando toda su potencia y versatilidad.

C# combina los mejores elementos de múltiples lenguajes de amplia difusión como C++, Java, Visual Basic o Delphi. De hecho, su creador Anders Heljsberg fue también el creador de muchos otros lenguajes y entornos como Turbo Pascal, Delphi o Visual J++. La idea principal detrás del lenguaje es combinar la potencia de lenguajes como C++ con la sencillez de lenguajes como Visual Basic, y que además la migración a este lenguaje por los programadores de C/C++/Java sea lo más inmediata posible.

Además de C#, Microsoft propociona Visual Studio.NET, la nueva versión de su entorno de desarrollo adaptada a la plataforma .NET y que ofrece una interfaz común para trabajar de manera cómoda y visual con cualquiera de los lenguajes de la plataforma .NET (por defecto, C++, C#, Visual Basic.NET y JScript.NET, aunque pueden añadirse nuevos lenguajes mediante los plugins que proporcionen sus fabricantes).

El propio autor de este artículo ha publicado un excelente libro de C# donde se puede ampliar toda esta información y, por supuesto, aprender el lenguaje. Consta de unas 260 páginas y ha sido escrito con la idea de que cualquiera con unos conocimientos mínimos de programación pueda seguirlo, aunque sería recomendable que el lector conociese C++, Java o lenguajes similares.

Puede ser descargado, junto con otra documentación, de la web del autor: <http://www.josanguapo.com/>

Informe de **José Antonio González Seco**

Mail: josanguapo@hotmail.com

URL: <http://www.josanguapo.com/>

Qué es .NET

Microsoft.NET es el conjunto de nuevas tecnologías en las que Microsoft ha estado trabajando durante los últimos años -y cuyo lanzamiento definitivo es inminente, estando ya disponible su primera versión Release Candidate- con los objetivos de:

- Mejorar sus sistemas operativos
- Mejorar su modelo de componentes COM+
- Obtener un entorno específicamente diseñado para el desarrollo y ejecución del software en forma de servicios que puedan ser tanto publicados como

accedidos a través de Internet de forma independiente del lenguaje de programación, modelo de objetos, sistema operativo y hardware utilizados tanto para desarrollarlos como para publicarlos. Éste entorno es lo que se denomina la plataforma.NET, y los servicios antes mencionados son a los que se denomina servicios web.

Para el desarrollo y ejecución de aplicaciones en este nuevo entorno tecnológico Microsoft proporciona el conjunto de herramientas conocido .NET Framework SDK, que es posible descargarlo gratuitamente de su sitio web <http://www.msdn.microsoft.com/net> e incluye compiladores de lenguajes como C#, Visual Basic.NET, Managed C++ y JScript.NET específicamente diseñados para crear aplicaciones para él.

El corazón de la plataforma.NET es el CLR (Common Language Runtime), que es una aplicación similar a un máquina virtual que se encarga de gestionar la ejecución de las aplicaciones para ella escritas. A estas aplicaciones les ofrece numerosos servicios que facilita su desarrollo y mantenimiento y favorecen su fiabilidad y seguridad. Entre ellos los principales son:

- Modelo de programación consistente y sencillo, completamente orientado a objetos.
- Eliminación del temido problema de compatibilidad entre DLLs conocido como "infierno de las DLLs"
- Ejecución multiplataforma
- Ejecución multilenguaje, hasta el punto de que es posible hacer cosas como capturar en un programa escrito en C# una excepción escrita en Visual Basic.NET que a su vez hereda de un tipo de excepción escrita en Cobol.NET. Aunque más arriba se ha dicho que en el .NET Framework sólo se ofrecen compiladores de C#, MC++, VB.NET y JScript.NET, lo cierto es que aparte Microsoft y terceros han -o están- desarrollado versiones adaptadas a .NET de muchísimos otros lenguajes como APL, CAML, Cobol, Eiffel, Fortran, Haskell, Java, Mercury, ML, Mondrian, Oberon, Oz, Pascal, Perl, Python, RPG, Scheme o Smalltalk
- Recolección de basura
- Aislamiento de memoria entre procesos y comprobaciones automáticas de seguridad de tipos en las conversiones
- Soporte multihilo
- Gestión del acceso a objetos remotos que permite el desarrollo de aplicaciones distribuidas de manera transparente a la ubicación real de cada uno de los objetos utilizados en las mismas.
- Seguridad avanzada, hasta el punto de que es posible limitar los permisos de ejecución del código en función de su procedencia (Internet, red local, CD-ROM, etc.), el usuario que lo ejecuta o la empresa que lo creó.
- Interoperabilidad con código preexistente, de manera que es posible utilizar con facilidad cualquier librería de funciones u objetos COM y COM+ creados con anterioridad a la aparición de la plataforma .NET
- Adecuación automática de la eficiencia de las aplicaciones a las características concretas de cada máquina donde se vaya a ejecutar

El propio autor de este artículo ha publicado un excelente libro donde podremos encontrar una descripción de .NET mucho más amplia. El libro en concreto trata sobre el lenguaje C#, desarrollado específicamente para la plataforma .NET.

Puede ser descargado, junto con otra documentación, de la web del autor: <http://www.josanguapo.com/>

Mail: josanguapo@hotmail.com
URL: <http://www.josanguapo.com/>

Qué es JSP

JSP es un acrónimo de Java Server Pages, que en castellano vendría a decir algo como Páginas de Servidor Java. Es, pues, una tecnología orientada a crear páginas web con programación en Java.

Bibliografía: Esta descripción de JSP está extraída de un PDF en inglés muy completo que introduce la tecnología, que se puede encontrar en la [página corporativa de Java de Sun Microsystems](#), en su [sección de aprendizaje online](#). A su vez, dicho manual proviene del [portal Java jGuru](#).

[jGuru: Introduction to JavaServer Pages technology](#)

Con JSP podemos crear aplicaciones web que se ejecuten en variados servidores web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML mezclado con etiquetas especiales para programar scripts de servidor en sintaxis Java. Por tanto, las JSP podremos escribirlas con nuestro editor HTML/XML habitual.

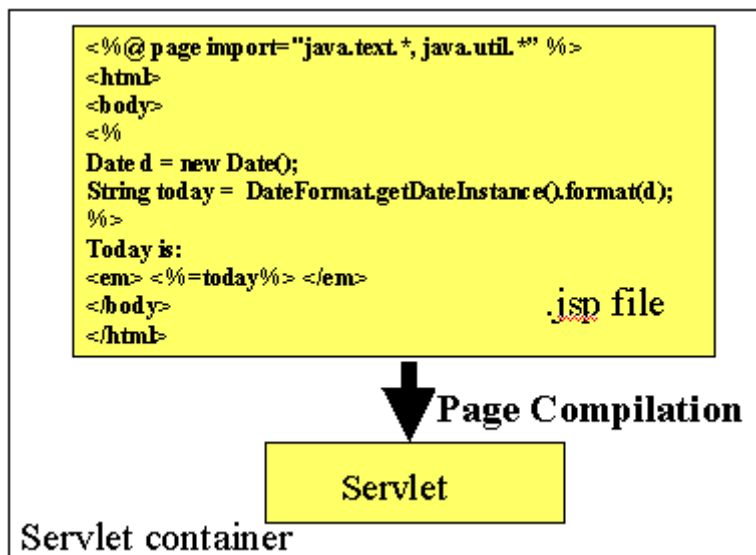
Motor JSP

El motor de las páginas JSP está basado en los servlets de Java -programas en Java destinados a ejecutarse en el servidor-, aunque el número de desarrolladores que pueden afrontar la programación de JSP es mucho mayor, dado que resulta mucho más sencillo aprender que los servlets.

En JSP creamos páginas de manera parecida a como se crean en [ASP](#) o [PHP](#) -otras dos [tecnologías de servidor](#)-. Generamos archivos con extensión .jsp que incluyen, dentro de la estructura de etiquetas HTML, las sentencias Java a ejecutar en el servidor. Antes de que sean funcionales los archivos, el motor JSP lleva a cabo una fase de traducción de esa página en un servlet, implementado en un archivo class (Byte codes de Java). Esta fase de traducción se lleva a cabo habitualmente cuando se recibe la primera solicitud de la página .jsp, aunque existe la opción de precompilar en código para evitar ese tiempo de espera la primera vez que un cliente solicita la página.

Ejemplo de página JSP

En la imagen siguiente se puede ver un ejemplo extremadamente simple de una página JSP y el esquema de conversión de esa página en un servlet.



Prerequisitos

Para aprender JSP, aparte de conocer HTML, será necesario comprender y tener algo de experiencia en la programación en [Java](#), que es un [lenguaje de programación Orientado a Objetos](#) por completo. Una vez conocida la programación en Java se puede estudiar por encima el sistema de Servlets, lo que nos dará una mejor idea del funcionamiento interno del motor JSP.

Para aprender Java podemos consultar algunos [enlaces del correspondiente directorio de nuestro buscador](#) de enlaces.

Además, necesitaremos descargar e instalar [Tomcat](#), el contenedor de servlets usado en la referencia oficial de implementación de JSP. Podemos acceder a un [ejercicio para aprender a realizar esta instalación](#), disponible también en la referencia de aprendizaje de la [página de Java](#).

Referencias JSP

Hemos creado una nueva [sección en nuestro directorio dedicada por completo a las páginas JSP](#), que será muy interesante para todo aquel que desee profundizar en el tema.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Qué es Oracle

Oracle es básicamente una herramienta cliente/servidor para la gestión de Bases de Datos. Es un producto vendido a nivel mundial, aunque la gran potencia que tiene y su elevado precio hace que sólo se vea en empresas muy grandes y multinacionales, por norma general. En el desarrollo de páginas web pasa lo mismo: como es un sistema muy caro no está tan extendido como otras bases de datos, por ejemplo, Access, MySQL, SQL Server, etc.

Vamos ahora en centrarnos en que es Oracle exactamente y como funciona la programación sobre éste. Oracle como antes he mencionado se basa en la tecnología cliente/servidor, pues bien, para su utilización primero sería necesario la instalación de la herramienta servidor (Oracle 8i) y posteriormente podríamos atacar a la base de datos desde otros equipos con herramientas de desarrollo como Oracle Designer y Oracle Developer, que son las herramientas básicas de programación sobre Oracle.

Para desarrollar en Oracle utilizamos PL/SQL un lenguaje de 5ª generación, bastante potente para tratar y gestionar la base de datos, también por norma general se suele utilizar SQL al crear un formulario.

Referencia: Podemos aprender [qué es el lenguaje SQL](#) en un artículo de DesarrolloWeb.com. Además, existe un [manual de SQL](#) para el que desee profundizar.

Es posible lógicamente atacar a la base de datos a través del SQL plus incorporado en el paquete de programas Oracle para poder realizar consultas, utilizando el lenguaje SQL.

El Developer es una herramienta que nos permite crear formularios en local, es decir, mediante esta herramienta nosotros podemos crear formularios, compilarlos y ejecutarlos, pero si queremos que los otros trabajen sobre este formulario deberemos copiarlo regularmente en una carpeta compartida para todos, de modo que, cuando quieran realizar un cambio, deberán copiarlo de dicha carpeta y luego volverlo a subir a la carpeta. Este sistema como podemos observar es bastante engorroso y poco fiable pues es bastante normal que las versiones se pierdan y se machaquen con frecuencia. La principal ventaja de esta herramienta es que es bastante intuitiva y dispone de un modo que nos permite componer el formulario, tal y como lo haríamos en Visual Basic o en Visual C, esto es muy de agradecer.

Los problemas anteriores quedan totalmente resueltos con Designer que es una herramienta que se conecta a la base de datos y por tanto creamos los formularios en ella, de esta manera todo el mundo se conecta mediante Designer a la aplicación que contiene todos los formularios y no hay problemas de diferentes versiones, esto es muy útil y perfecto para evitar machacar el trabajo de otros. Pero el principal y más notable problema es la falta de un entorno visual para diseñar el formulario, es decir, nos aparece una estructura como de árbol en la cual insertamos un formulario, a la vez dentro de éste insertamos bloques o módulos que son las estructuras que contendrán los elementos del formularios, que pueden estar basados en tablas o no.

Por lo tanto si queremos hacer formularios para practicar o para probar qué es esto de Oracle, os recomiendo que uséis Developer pues es mucho más fácil e intuitivo al principio.

Informe de **David Masip**
Mail: d.masip@barcelo.com

Qué es ActiveX

ActiveX es una tecnología de Microsoft para el desarrollo de páginas dinámicas. Tiene presencia en la programación del lado del servidor y del lado del cliente, aunque existan diferencias en el uso en cada uno de esos dos casos.

En el cliente:

Son pequeños programas que se pueden incluir dentro de páginas web y sirven para realizar acciones de diversa índole. Por ejemplo hay controles ActiveX para mostrar un calendario, para implementar un sistema de FTP, etc.

Son un poco parecidos a los [Applets de Java](#) en su funcionamiento, aunque una diferencia fundamental es la seguridad, pues un Applet de Java no podrá tomar privilegios para realizar acciones malignas (como borrar el disco duro) y los controles ActiveX sí que pueden otorgarse permisos para hacer cualquier cosa.

Los controles ActiveX son particulares de Internet Explorer.

En el servidor

También existen controles ActiveX del servidor y la gente que conozca ASP seguro que los utiliza ya, aunque sea sin darse cuenta. Por ejemplo, cuando realizamos una conexión con una base de datos, estamos utilizando un control ActiveX del servidor.

Para aprender más de ActiveX de servidor, sería interesante consultar el artículo [Componentes de servidor ASP](#)

Desarrollo de ActiveX

Los controles ActiveX se desarrollan con entornos de Microsoft para la creación de aplicaciones Windows, como pueden ser Visual Basic Script o Visual C. Se nos escapa totalmente de este artículo el explicar algo del método de desarrollo, pero lo que sí cabe señalar es que existen muchos controles ActiveX tanto del lado del servidor como del cliente, que están ya desarrollados y podemos incluirlos fácilmente en nuestras creaciones.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

TUTORIAL SOBRE SQL

Qué es SQL

Las aplicaciones en red son cada día más numerosas y versátiles. En muchos casos, el esquema básico de operación es una serie de scripts que rigen el comportamiento de una base de datos.

Debido a la diversidad de lenguajes y de bases de datos existentes, la manera de comunicar entre unos y otras sería realmente complicada a gestionar de no ser por la existencia de estándares que nos permiten el realizar las operaciones básicas de una forma universal.

Es de eso de lo que trata el Structured Query Language que no es mas que un lenguaje estándar de comunicación con bases de datos. Hablamos por tanto de un lenguaje normalizado que nos permite trabajar con cualquier tipo de lenguaje (ASP o PHP) en combinación con cualquier tipo de base de datos (MS Access, SQL Server, MySQL...).

El hecho de que sea estándar no quiere decir que sea idéntico para cada base de datos. En efecto, determinadas bases de datos implementan funciones específicas que no tienen necesariamente que funcionar en otras.

Aparte de esta universalidad, el SQL posee otras dos características muy apreciadas. Por una parte, presenta una potencia y versatilidad notables que contrasta, por otra, con su accesibilidad de aprendizaje.

[El manual de SQL de desarrolloweb](#) pretende dar a conocer las operaciones básicas que se pueden realizar con SQL y que tienen una aplicación directa con la creación de aplicaciones en red sin profundizar más de lo estrictamente necesario. Buscamos con ello ofrecer al webmaster un manual de referencia práctico y aplicado.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Tipos de campo

Como sabemos una base de datos esta compuesta de tablas donde almacenamos registros catalogados en función de distintos campos (características).

Un aspecto previo a considerar es la naturaleza de los valores que introducimos en esos campos. Dado que una base de datos trabaja con todo tipo de informaciones, es importante especificarle qué tipo de valor le estamos introduciendo de manera a, por un lado, facilitar la búsqueda posteriormente y por otro, optimizar los recursos de memoria.

Cada base de datos introduce tipos de valores de campo que no necesariamente están presentes en otras. Sin embargo, existe un conjunto de tipos que están representados en la totalidad de estas bases. Estos tipos comunes son los siguientes:

Alfanuméricos	Contienen cifras y letras. Presentan una longitud limitada
----------------------	--

	(255 caracteres)
Numéricos	Existen de varios tipos, principalmente, enteros (sin decimales) y reales (con decimales).
Booleanos	Poseen dos formas: Verdadero y falso (Sí o No)
Fechas	Almacenan fechas facilitando posteriormente su explotación. Almacenar fechas de esta forma posibilita ordenar los registros por fechas o calcular los días entre una fecha y otra...
Memos	Son campos alfanuméricos de longitud ilimitada. Presentan el inconveniente de no poder ser indexados (veremos más adelante lo que esto quiere decir).
Autoincrementables	Son campos numéricos enteros que incrementan en una unidad su valor para cada registro incorporado. Su utilidad resulta más que evidente: Servir de identificador ya que resultan exclusivos de un registro.

Informe de **Rubén Alvarez**
 Mail: ruben@desarrolloweb.com

Añadir un nuevo registro

Los registros pueden ser introducidos a partir de sentencias que emplean la instrucción Insert.

La sintaxis utilizada es la siguiente:

Insert Into nombre_tabla (nombre_campo1, nombre_campo2,...) Values (valor_campo1, valor_campo2...)

Un ejemplo sencillo a partir de nuestra tabla modelo es la introducción de un nuevo cliente lo cual se haría con una instrucción de este tipo:

Insert Into clientes (nombre, apellidos, direccion, poblacion, codigopostal, email, pedidos) Values ('Perico', 'Palotes', 'Percebe nº13', 'Lepe', '123456', 'perico@desarrolloweb.com', 33)

Como puede verse, los campos no numéricos o booleanos van delimitados por apostrofes: '. También resulta interesante ver que el código postal lo hemos guardado como un campo no numérico. Esto es debido a que en determinados países (Inglaterra, como no) los códigos postales contienen también letras.

Nota: Si deseamos practicar con una base de datos que está vacía primero debemos crear las tablas que vamos a llenar. Las tablas también se crean con sentencias SQL y [aprendemos a hacerlo en el último capítulo](#).

Aunque, de todos modos, puede que sea más cómodo utilizar un programa con interfaz gráfica, como Access, que nos puede servir para crear las tablas en bases de datos del propio [Access](#) o por ODBC a otras bases de datos como [SQL Server](#) o [MySQL](#), por poner dos ejemplos.

Otra posibilidad en una base de datos como MySQL, sería crear las tablas utilizando un software como [PhpMyAdmin](#).

Por supuesto, no es imprescindible rellenar todos los campos del registro. Eso sí, puede ser que determinados campos sean necesarios. Estos campos necesarios pueden ser definidos cuando construimos nuestra tabla mediante la base de datos.

Nota: Si no insertamos uno de los campos en la base de datos se inicializará con el valor por defecto que hayamos definido a la hora de crear la tabla. Si no hay valor por defecto, probablemente se inicialice como NULL (vacío), en caso de que este campo permita valores nulos. Si ese campo no permite valores nulos (eso se define también al crear la tabla) lo más seguro es que la ejecución de la sentencia SQL nos de un error.

Resulta muy interesante, ya veremos más adelante el por qué, el introducir durante la creación de nuestra tabla un campo autoincrementable que nos permita asignar un único número a cada uno de los registros. De este modo, nuestra tabla clientes presentaría para cada registro un número exclusivo del cliente el cual nos será muy útil cuando consultemos varias tablas simultáneamente.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Borrar un registro

Para borrar un registro nos servimos de la instrucción Delete. En este caso debemos especificar cual o cuales son los registros que queremos borrar. Es por ello necesario establecer una selección que se llevara a cabo mediante la cláusula Where.

La forma de seleccionar se verá detalladamente en capítulos posteriores. Por ahora nos contentaremos de mostrar cuál es el tipo de sintaxis utilizado para efectuar estas supresiones:

```
Delete From nombre_tabla Where condiciones_de_selección
```

Nota: Si deseamos practicar con una base de datos que está vacía primero debemos crear las tablas que vamos a llenar. Las tablas también se crean con sentencias SQL y [aprendemos a hacerlo en el último capítulo](#).

Si queremos por ejemplo borrar todos los registros de los clientes que se llamen Perico lo haríamos del siguiente modo:

```
Delete From clientes Where nombre='Perico'
```

Hay que tener cuidado con esta instrucción ya que si no especificamos una condición con Where, lo que estamos haciendo es **borrar toda la tabla**:

Delete From clientes

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Actualizar un registro

Update es la instrucción que nos sirve para modificar nuestros registros. Como para el caso de Delete, necesitamos especificar por medio de Where cuáles son los registros en los que queremos hacer efectivas nuestras modificaciones. Además,

obviamente, tendremos que especificar cuáles son los nuevos valores de los campos que deseamos actualizar. La sintaxis es de este tipo:

```
Update nombre_tabla Set nombre_campo1 = valor_campo1, nombre_campo2 = valor_campo2,... Where condiciones_de_selección
```

Un ejemplo aplicado:

```
Update clientes Set nombre='José' Where nombre='Pepe'
```

Mediante esta sentencia cambiamos el nombre Pepe por el de José en todos los registros cuyo nombre sea Pepe.

Aquí también hay que ser cuidadoso de no olvidarse de usar Where, de lo contrario, modificaríamos todos los registros de nuestra tabla.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Selección de tablas I

La selección total o parcial de una tabla se lleva a cabo mediante la instrucción Select. En dicha selección hay que especificar:

- Los campos que queremos seleccionar
- La tabla en la que hacemos la selección

En nuestra tabla modelo de clientes podríamos hacer por ejemplo una selección del nombre y dirección de los clientes con una instrucción de este tipo:

```
Select nombre, dirección From clientes
```

Si quisiésemos seleccionar todos los campos, es decir, **toda la tabla**, podríamos utilizar el comodín * del siguiente modo:

Select * From clientes

Resulta también muy útil el filtrar los registros mediante condiciones que vienen expresadas después de la **cláusula Where**. Si quisiésemos mostrar los clientes de una determinada ciudad usaríamos una expresión como esta:

```
Select * From clientes Where poblacion Like 'Madrid'
```

Además, podríamos **ordenar los resultados** en función de uno o varios de sus campos. Para este último ejemplo los podríamos ordenar por nombre así:

```
Select * From clientes Where poblacion Like 'Madrid' Order By nombre
```

Teniendo en cuenta que puede haber más de un cliente con el mismo nombre, podríamos dar un segundo criterio que podría ser el apellido:

```
Select * From clientes Where poblacion Like 'Madrid' Order By nombre, apellido
```

Si invirtiésemos el orden « nombre,apellido » por « apellido, nombre », el resultado sería distinto. Tendríamos los clientes ordenados por apellido y aquellos que

tuviesen apellidos idénticos se subclasificarían por el nombre.

Es posible también **clasificar por orden inverso**. Si por ejemplo quisiésemos ver nuestros clientes por orden de pedidos realizados teniendo a los mayores en primer lugar escribiríamos algo así:

```
Select * From clientes Order By pedidos Desc
```

Una opción interesante es la de efectuar **selecciones sin coincidencia**. Si por ejemplo buscásemos el saber en qué ciudades se encuentran nuestros clientes sin necesidad de que para ello aparezca varias veces la misma ciudad usaríamos una sentencia de esta clase:

```
Select Distinct poblacion From clientes Order By poblacion
```

Así evitaríamos ver repetido Madrid tantas veces como clientes tengamos en esa población.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Selección de tablas II

Hemos querido compilar a modo de tabla ciertos operadores que pueden resultar útiles en determinados casos. Estos operadores serán utilizados después de la cláusula Where y pueden ser **combinados hábilmente mediante paréntesis** para optimizar nuestra selección a muy altos niveles.

Operadores matemáticos:	
>	Mayor que
<	Menor que
>=	Mayor o igual que
<=	Menor o igual que
<>	Distinto
=	Igual

Operadores lógicos
And
Or
Not

Otros operadores	
Like	Selecciona los registros cuyo valor de campo se asemeje, no teniendo en cuenta mayúsculas y minúsculas.
In y Not In	Da un conjunto de valores para un campo para los cuales la condición de selección es (o no) valida

Is Null y Is Not Null	Selecciona aquellos registros donde el campo especificado esta (o no) vacío.
Between...And	Selecciona los registros comprendidos en un intervalo
Distinct	Selecciona los registros no coincidentes
Desc	Clasifica los registros por orden inverso

Comodines	
*	Sustituye a todos los campos
%	Sustituye a cualquier cosa o nada dentro de una cadena
_	Sustituye un solo carácter dentro de una cadena

Veamos a continuación aplicaciones practicas de estos operadores.

En esta sentencia seleccionamos todos los clientes de Madrid cuyo nombre no es Pepe. Como puede verse, empleamos **Like** en lugar de = simplemente para evitar inconvenientes debido al empleo o no de mayúsculas.

Select * From clientes Where poblacion **Like** 'madrid' **And Not** nombre **Like** 'Pepe'

Si quisiéramos recoger en una selección a los clientes de nuestra tabla cuyo **apellido comienza por A y cuyo número de pedidos esta comprendido entre 20 y 40:**

Select * From clientes Where apellidos **like** 'A%' And pedidos **Between 20 And 40**


El operador **In**, lo veremos más adelante, es muy práctico para consultas en varias tablas. Para casos en una sola tabla es empleado del siguiente modo:

Select * From clientes Where poblacion **In ('Madrid','Barcelona','Valencia')**

De esta forma **seleccionamos aquellos clientes que vivan en esas tres ciudades.**

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Selección de tablas III

 Una base de datos puede ser considerada como un conjunto de tablas. Estas tablas en muchos casos están relacionadas entre ellas y se complementan unas con otras.

Refiriéndonos a nuestro clásico ejemplo de una base de datos para una aplicación de e-comercio, la tabla clientes de la que hemos estado hablando puede estar perfectamente coordinada con una tabla donde almacenamos los pedidos realizados por cada cliente. Esta tabla de pedidos puede a su vez estar conectada con una tabla donde almacenamos los datos correspondientes a cada artículo del inventario.

De este modo podríamos fácilmente obtener informaciones contenidas en esas tres tablas como puede ser la designación del artículo más popular en una determinada región donde la designación del artículo sería obtenida de la tabla de artículos, la popularidad (cantidad de veces que ese artículo ha sido vendido) vendría de la tabla de pedidos y la región estaría comprendida obviamente en la tabla clientes.

Este tipo de organización basada en múltiples tablas conectadas nos permite trabajar con tablas mucho más manejables a la vez que nos evita copiar el mismo campo en varios sitios ya que podemos acceder a él a partir de una simple llamada a la tabla que lo contiene.

En este capítulo veremos como, sirviéndonos de lo aprendido hasta ahora, podemos realizar fácilmente selecciones sobre varias tablas. Definamos antes de nada las diferentes tablas y campos que vamos a utilizar en nuestros ejemplos:

Tabla de clientes	
Nombre campo	Tipo campo
id_cliente	Numérico entero
nombre	Texto
apellidos	Texto
direccion	Texto
poblacion	Texto
codigopostal	Texto
telefono	Numérico entero
email	Texto

Tabla de pedidos	
Nombre campo	Tipo campo
id_pedido	Numérico entero
id_cliente	Numérico entero
id_articulo	Numérico entero
fecha	Fecha
cantidad	Numérico entero

Tabla de artículos	
Nombre campo	Tipo campo
id_articulo	Numérico entero
titulo	Alfanumérico
autor	Alfanumérico
editorial	Alfanumérico
precio	Numérico real

Estas tablas pueden ser utilizadas simultáneamente para extraer informaciones de todo tipo. Supongamos que queremos enviar un mailing a todos aquellos que hayan realizado un pedido ese mismo día. Podríamos escribir algo así:

Select clientes.apellidos, clientes.email From clientes,pedidos Where pedidos.fecha like '25/02/00' And pedidos.id_cliente= clientes.id_cliente

Como puede verse esta vez, después de la cláusula From, introducimos el nombre de las dos tablas de donde sacamos las informaciones. Además, el nombre de cada campo va precedido de la tabla de proveniencia separados ambos por un punto. En los campos que poseen un nombre que solo aparece en una de las tablas, no es necesario especificar su origen aunque a la hora de leer la sentencia puede resultar más claro el precisarlo. En este caso el campo fecha podría haber sido designado como "fecha" en lugar de "pedidos.fecha".

Veamos otro ejemplo más para consolidar estos nuevos conceptos. Esta vez queremos ver el título del libro correspondiente a cada uno de los pedidos realizados:

Select pedidos.id_pedido, articulos.titulo From pedidos, articulos Where pedidos.id_articulo=articulos.id_articulo

En realidad la filosofía continua siendo la misma que para la consulta de una única tabla.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Selección de tablas IV

Además de los criterios hasta ahora explicados para realizar las consultas en tablas, SQL permite también aplicar un conjunto de funciones predefinidas. Estas funciones, aunque básicas, pueden ayudarnos en algunos momentos a expresar nuestra selección de una manera más simple sin tener que recurrir a operaciones adicionales por parte del script que estemos ejecutando.

Algunas de estas funciones son representadas en la tabla siguiente :

Función	Descripción
Sum(campo)	Calcula la suma de los registros del campo especificado
Avg(Campo)	Calcula la media de los registros del campo especificado
Count(*)	Nos proporciona el valor del numero de registros que han sido seleccionados
Max(Campo)	Nos indica cual es el valor máximo del campo
Min(Campo)	Nos indica cual es el valor mínimo del campo

Dado que el campo de la función no existe en la base de datos, sino que lo estamos generando virtualmente, esto puede crear inconvenientes cuando estamos trabajando con nuestros scripts a la hora de tratar su valor y su nombre de campo. Es por ello que el valor de la **función ha de ser recuperada a partir de un alias** que nosotros especificaremos en la sentencia SQL a partir de la instrucción **AS**. La

cosa podría quedar así:

Select Sum(total) As suma_pedidos From pedidos

A partir de esta sentencia calculamos la suma de los valores de todos los pedidos realizados y almacenamos ese valor en un campo virtual llamado suma_pedidos que podrá ser utilizado como cualquier otro campo por nuestras paginas dinámicas.

Por supuesto, todo lo visto hasta ahora puede ser aplicado en este tipo de funciones de modo que, por ejemplo, podemos establecer condiciones con la cláusula Where construyendo sentencias como esta:

Select Sum(cantidad) as suma_articulos From pedidos Where id_articulo=6

Esto nos proporcionaría la cantidad de **ejemplares de un determinado libro que han sido vendidos**.

Otra propiedad interesante de estas funciones es que **permiten realizar operaciones con varios campos dentro de un mismo paréntesis**:

Select Avg(total/cantidad) From pedidos

Esta sentencia da como resultado el **precio medio al que se están vendiendo los libros**. Este resultado no tiene por qué coincidir con el del **precio medio de los libros presentes en el inventario**, ya que, puede ser que la gente tenga tendencia a comprar los libros caros o los baratos:

Select Avg(precio) as precio_venta From articulos

Una cláusula interesante en el uso de funciones es Group By. Esta cláusula nos permite agrupar registros a los cuales vamos a aplicar la función. Podemos por ejemplo calcular el **dinero gastado por cada cliente**:

Select id_cliente, Sum(total) as suma_pedidos From pedidos Group By id_cliente

O saber el **numero de pedidos que han realizado**:

Select id_cliente, Count(*) as numero_pedidos From pedidos Group By id_cliente

Las posibilidades como vemos son numerosas y pueden resultar prácticas. Todo queda ahora a disposición de nuestras ocurrencias e imaginación.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Optimizar prestaciones I

Las bases de datos (BD) cuanto más extensas requieren una mayor atención a la hora de organizar sus contenidos. Cuando se trabaja con tablas de miles o decenas

de miles de registros la búsqueda de un determinado dato puede resultar un proceso largo que ralentiza enormemente la creación de nuestra página.

Es por ello importante tener en cuenta una serie de aspectos indispensables para el mejor funcionamiento de la base.

Gestión y elección de los índices

Los índices son campos elegidos arbitrariamente por el constructor de la BD que permiten la búsqueda a partir de dicho campo a una velocidad notablemente superior. Sin embargo, esta ventaja se ve contrarrestada por el hecho de ocupar mucha más memoria (el doble más o menos) y de requerir para su inserción y actualización un tiempo de proceso superior.

Evidentemente, **no podemos indexar todos los campos** de una tabla extensa ya que doblamos el tamaño de la BD. Igualmente, tampoco sirve de mucho el indexar todos los campos en una tabla pequeña ya que las selecciones pueden efectuarse rápidamente de todos modos.

Un caso en el que los índices pueden resultar muy útiles es cuando realizamos peticiones simultáneas sobre varias tablas. En este caso, el proceso de selección puede acelerarse sensiblemente si **indexamos los campos que sirven de nexo entre las dos tablas**. En el ejemplo de nuestra librería virtual estos campos serían `id_cliente` e `id_articulo`.

Los índices pueden resultar contraproducentes si los introducimos sobre campos triviales a partir de los cuales no se realiza ningún tipo de petición ya que, además del problema de memoria ya mencionado, estamos ralentizando otras tareas de la base de datos como son la edición, inserción y borrado. Es por ello que vale la pena pensárselo dos veces antes de indexar un campo que no sirve de criterio para búsquedas de los internautas y que es usado con muy poca frecuencia por razones de mantenimiento.

Gestión de los nexos entre tablas

El enlace entre tablas es uno de los puntos más peliagudos y que puede llevar a la absoluta ralentización de la base de datos a causa "pequeños" detalles que resultan ser fatales.

Imaginemos que trabajamos con una pequeña BD constituida por dos tablas de 1000 registros cada una. Imaginemos ahora una selección simultánea en la que imponemos la condición de que el valor un campo de la primera sea igual a de una segunda, algo que se realiza con mucha frecuencia. En este tipo de casos, la BD leerá y comparará cada valor de campo de una con cada valor de campo de la otra. Esto representaría un millón de lecturas. Este hecho podría agravarse si consultamos una tercera tabla al mismo tiempo y podría llegar a ser catastrófico si tenemos en cuenta que la BD esta siendo consultada por varios internautas al mismo tiempo.

Para evitar situaciones de colapso, es necesario **indexar cada uno de los campos que sirven de enlace entre esas tablas**. En el ejemplo de nuestra librería virtual, ya lo hemos dicho, estos campos serían `id_cliente` e `id_articulo`. Además, resulta también de vital importancia el **definir esos campos de una forma estrictamente idéntica en cada una de las tablas**, es decir, el campo ha de ser de la misma naturaleza y características. No vale definirlo como `real` en una tabla y `entero` en otra o cambiar la longitud máxima para los alfanuméricos o que en una tabla sea de longitud constante y en otra variable...

El gestionar inteligentemente estos aspectos puede solucionarnos muchos quebraderos de cabeza y permitir a los internautas navegar más agradablemente por nuestro sitio.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Optimizar prestaciones II

Gestion de los campos

Ya hemos comentado por encima los diferentes tipos de campo existentes en una base de datos. La elección del tipo de campo apropiado para cada caso puede ayudarnos también a optimizar el tamaño y rapidez de nuestra base de datos.

Las preguntas que hay que hacerse a la hora de elegir la naturaleza y dimensiones del campo son:

-¿Qué tipo de dato voy a almacenar en el campo? Números, texto, fechas...

-¿Cuál es el tamaño máximo que espero que pueda alcanzar alguno de los registros del campo?

Hay que tener en cuenta que cuanto más margen le demos al valor máximo del campo, más aumentará el tamaño de nuestra base de datos y más tiempo tardará en realizar las consultas. Además, el factor tamaño puede verse agravado si estamos definiendo un campo indexado, para los cuales, el espacio ocupado es aproximadamente del doble.

Un consejo práctico es que las fechas sean almacenadas en formato de fecha ya que ello nos permite reducir el espacio que ocupan en memoria de más del doble y por otro lado, podremos aprovechar las prestaciones que SQL y nuestro lenguaje de servidor nos ofrecen. Podremos calcular la diferencia de días entre dos fechas, ordenar los registros por fecha, mostrar los registros comprendidos en un intervalo de tiempo...

Existe la posibilidad para los campos de texto de fijar una cierta longitud para el campo o dejar que cada registro tenga una longitud variable en función del número de caracteres que posea. Elegir campos de longitud variable nos puede ayudar a optimizar los recursos de memoria de la BD, no obstante, es un arma de doble filo ya que las consultas se realizan más lentamente puesto que obligamos a la tabla a establecer cuál es el tamaño de cada registro que se está comparando en lugar de saberlo de antemano. Es por tanto aconsejable, para los campos indexados de pequeño tamaño, atribuirles una longitud fija.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Algunos trucos prácticos

Eliminar llamadas a bases de datos

En páginas tipo portal en las que a los lados se encuentran enlaces que son impresos a partir de bases de datos (distintas secciones, servicios,...) existe siempre un efecto ralentizador debido a que se trata de páginas altamente visitadas que efectúan múltiples llamadas a BD sistemáticamente en cada una de sus páginas.

Una forma de agilizar la visualización de estas páginas es textualizando estos enlaces a partir de scripts internos. Pongamos el ejemplo de Desarrolloweb:

Como puede verse, a los lados hay secciones como "Vuestras páginas", "Cosecha del 2000", "Manuales" cuyos enlaces están almacenados en bases de datos. Sin embargo, los enlaces que se visualizan en la página no han sido obtenidos por llamadas a bases de datos sino que, cada vez que un nuevo elemento de la sección es añadido, esto actualiza automáticamente, por medio de un script, un archivo texto en el que el nuevo enlace es incluido y él más antiguo es eliminado. Es, de hecho, este archivo texto el que es insertado en el código fuente de la página. De este modo evitamos media docena de llamadas a bases de datos cada vez que una página es vista lo cual permite optimizar recursos de servidor de una manera significativa.

Eliminar palabras cortas y repeticiones

En situaciones en la que nuestra base de datos tiene que almacenar campos de texto extremadamente largos y dichos campos son requeridos para realizar selecciones del tipo LIKE '%algo%', los recursos de la BD pueden verse sensiblemente mermados. Una forma de ayudar a gestionar este tipo búsquedas es incluyendo un campo adicional.

Este campo adicional puede ser creado automáticamente por medio de scripts y en él incluiríamos el texto original, del cual habremos eliminado palabras triviales como artículos, preposiciones o posesivos. Nos encargaremos además de eliminar las palabras que estén repetidas. De esta forma podremos disminuir sensiblemente el tamaño del campo que va a ser realmente consultado.

Hemos comentado en otros capítulos que los campos texto de mas de 255 caracteres denominados memo no pueden ser indexados. Si aún después de esta primera filtración nuestro campo continua siendo demasiado largo para ser indexado, lo que se puede hacer es cortarlo en trozos de 255 caracteres de manera a que lo almacenemos en distintos campos que podrán ser indexados y por tanto consultados con mayor rapidez.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Creación de tablas

En general, la mayoría de las bases de datos poseen potentes editores de bases que permiten la creación rápida y sencilla de cualquier tipo de tabla con cualquier tipo de formato.

Sin embargo, una vez la base de datos está alojada en el servidor, puede darse el caso de que queramos introducir una nueva tabla ya sea con carácter temporal (para gestionar un carrito de compra por ejemplo) o bien permanente por necesidades concretas de nuestra aplicación.

En estos casos, podemos, a partir de una sentencia SQL, crear la tabla con el formato que deseemos lo cual nos puede ahorrar más de un quebradero de cabeza.

Este tipo de sentencias son especialmente útiles para bases de datos como Mysql, las cuales trabajan directamente con comandos SQL y no por medio de editores.

Para crear una tabla debemos especificar diversos datos: El nombre que le queremos asignar, los nombres de los campos y sus características. Además, puede ser necesario especificar cuáles de estos campos van a ser índices y de qué tipo van a serlo.

La sintaxis de creación puede variar ligeramente de una base de datos a otra ya que los tipos de campo aceptados no están completamente estandarizados.

A continuación os explicamos someramente la sintaxis de esta sentencia y os proponemos una serie de ejemplos prácticos:

Sintaxis

```
Create Table nombre_tabla
(
nombre_campo_1 tipo_1
nombre_campo_2 tipo_2
nombre_campo_n tipo_n
Key(campo_x,...)
)
```

Pongamos ahora como ejemplo la creación de la tabla pedidos que hemos empleado en capítulos previos:

```
Create Table pedidos
(
id_pedido INT(4) NOT NULL AUTO_INCREMENT,
id_cliente INT(4) NOT NULL,
id_articulo INT(4)NOT NULL,
fecha DATE,
cantidad INT(4),
total INT(4), KEY(id_pedido,id_cliente,id_articulo)
)
```

En este caso creamos los campos *id* los cuales son considerados de tipo entero de una longitud especificada por el número entre paréntesis. Para *id_pedido* requerimos que dicho campo se incremente automáticamente (AUTO_INCREMENT) de una unidad a cada introducción de un nuevo registro para, de esta forma, automatizar su creación. Por otra parte, para evitar un mensaje de error, es necesario requerir que los campos que van a ser definidos como índices no puedan ser nulos (NOT NULL).

El campo *fecha* es almacenado con formato de fecha (DATE) para permitir su correcta explotación a partir de las funciones previstas a tal efecto.

Finalmente, definimos los índices enumerándolos entre paréntesis precedidos de la palabra KEY o INDEX.

Del mismo modo podríamos crear la tabla de *artículos* con una sentencia como ésta:

```

Create Table articulos
(
id_articulo INT(4) NOT NULL AUTO_INCREMENT,
titulo VARCHAR(50),
autor VARCHAR(25),
editorial VARCHAR(25),
precio REAL,
KEY(id_articulo)
)

```

En este caso puede verse que los campos alfanuméricos son introducidos de la misma forma que los numéricos. Volvemos a recordar que en tablas que tienen campos comunes es de vital importancia definir estos campos de la misma forma para el buen funcionamiento de la base.

Muchas son las opciones que se ofrecen al generar tablas. No vamos a tratarlas detalladamente pues sale de lo estrictamente práctico. Tan sólo mostraremos algunos de los tipos de campos que pueden ser empleados en la creación de tablas con sus características:

Tipo	Bytes	Descripción
INT o INTEGER	4	Números enteros. Existen otros tipos de mayor o menor longitud específicos de cada base de datos.
DOUBLE o REAL	8	Números reales (grandes y con decimales). Permiten almacenar todo tipo de número no entero.
CHAR	1/caracter	Alfanuméricos de longitud fija predefinida
VARCHAR	1/caracter+1	Alfanuméricos de longitud variable
DATE	3	Fechas, existen múltiples formatos específicos de cada base de datos
BLOB	1/caracter+2	Grandes textos no indexables
BIT o BOOLEAN	1	Almacenan un bit de información (verdadero o falso)

Informe de **Rubén Alvarez**
 Mail: ruben@desarrolloweb.com

Internet Information Server (IIS) es el servidor de páginas web avanzado de la plataforma Windows. Se distribuye gratuitamente junto con las versiones de Windows basadas en NT, como pueden ser Windows 2000 Profesional o Windows 2000 Server, así como Windows XP, también en sus versiones Profesional y Server.

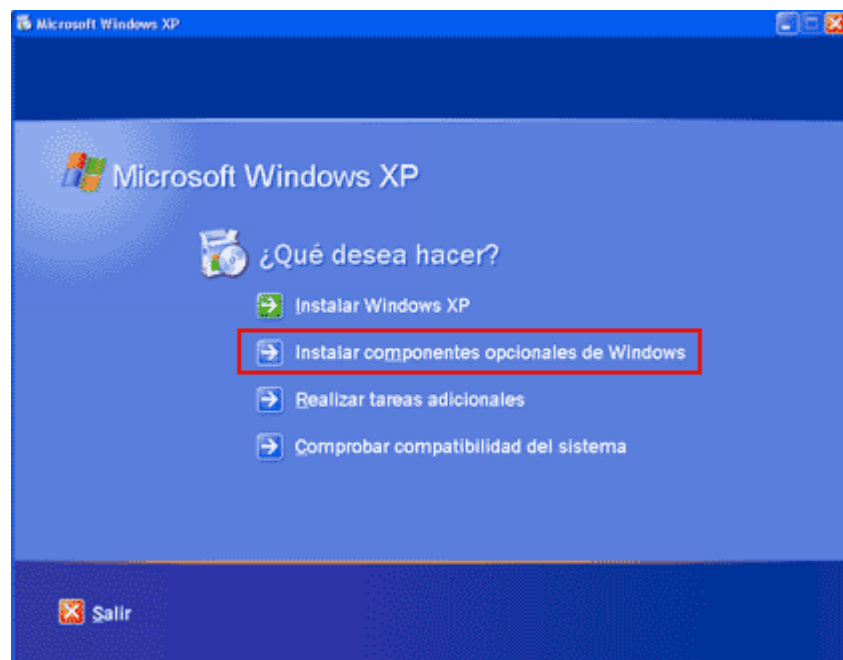
Estas normas de instalación son aplicables, a nivel general, a las que podemos encontrarnos en las distintas versiones de los sistemas operativos comentados antes, si bien hemos tomado Windows XP profesional para relatar los pasos y tomar las imágenes de las pantallas.

Nota: Windows 95, 98, las versiones Home, de Windows XP, y ME, de Windows 2000, no se admite la instalación de IIS. En su lugar podemos probar a instalar el Personal Web Server, que se explica en el artículo [Instalación de Personal Web Server](#).

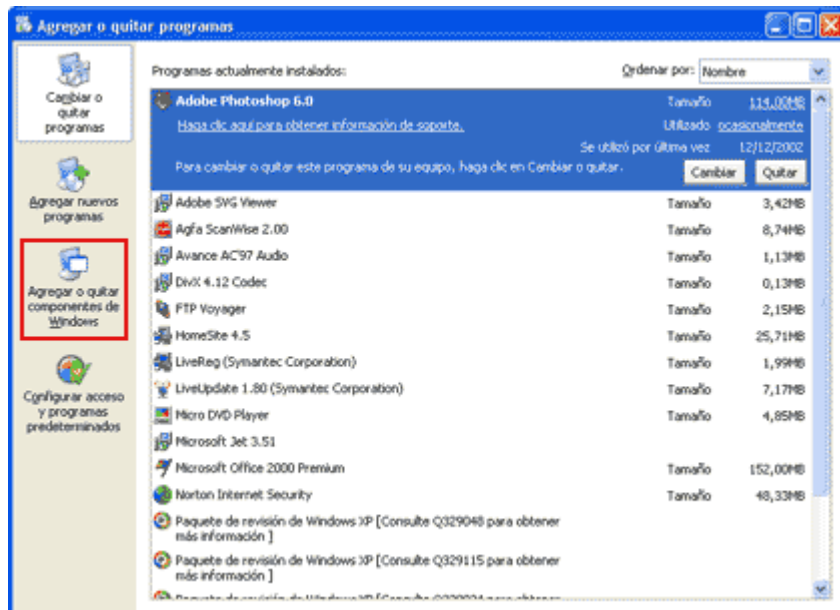
Agregar componentes adicionales de Windows

IIS se puede encontrar en el propio CD de instalación de Windows XP Profesional. Hay que acceder a la opción de "Instalar componentes opcionales de Windows" para poder cargarlo en nuestro sistema. Para ello tenemos dos opciones:

1) Insertar el CD de instalación de Windows y en la ventana de autoarranque que se muestra, seleccionar la opción que pone "Instalar componentes opcionales de Windows"

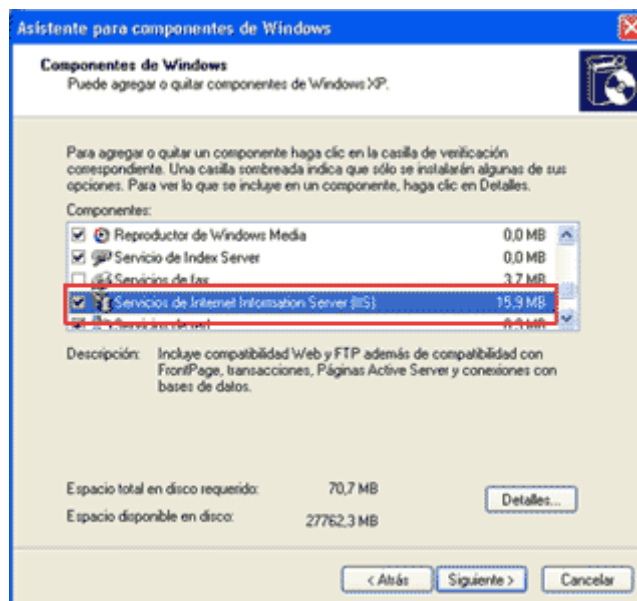


2) En el Panel de control, seleccionar la opción de "Agregar o quitar programas" y en la ventana que sale, pulsar sobre el icono de la izquierda marcado como "Seleccionar o quitar componentes de Windows".



Ahora nos muestra la ventana para seleccionar los componentes adicionales de Windows que hay disponibles. En la lista, marcamos la opción "Servicios de Internet Information Server (IIS)". Por defecto se seleccionan unos cuantos componentes, dentro de los que ofrece la instalación de IIS. Nosotros podemos elegir qué componentes deseamos instalar apretando el botón marcado como "Detalles". Entre los componentes posibles se encuentran las extensiones de Frontpage, documentación, servicios adicionales de IIS, un servidor de FTP (para la transferencia de ficheros con el servidor por FTP), incluso uno de SMTP (para el envío de correos electrónicos).

Si no sabemos qué componentes instalar podemos dejar las opciones como aparecen en un principio, pues para la mayoría de los casos serán válidas. Sólo un detalle: puede ser adecuado no instalar las extensiones de Frontpage en caso de que no pensemos que se vayan a utilizar.



Una vez hemos instalado los componentes deseados, podemos y apretar el botón

de "Siguiente" para comenzar la instalación, que se alargará unos minutos.

Acceder al servidor web

Podemos acceder al servidor web para comprobar si se ha instalado correctamente IIS. Para ello simplemente debemos escribir `http://localhost` en Internet Explorer y debería aparecer una página web informando que IIS está correctamente instalado. Además, aparecerá la documentación de IIS en una ventana emergente, si es que fue instalada.



Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Sitio web predeterminado en IIS

Lo que se muestra cuando accedemos a `http://localhost` es el sitio web predeterminado, que se guarda en nuestro disco duro, concretamente en la carpeta `C:\inetpub\wwwroot`

Si accedemos a dicha carpeta desde Mi PC podremos ver los archivos que se están sirviendo como sitio web predeterminado. Encontraremos, entre otros archivos uno llamado "iisstart.asp" que es el que se pone en marcha al acceder a este directorio.

Colocar nuestras propias páginas

Lo lógico ahora es que deseemos colocar nuestras propias páginas web para que las sirva IIS. Si nuestro sitio web es bastante simple podríamos colocar todos los archivos dentro de la carpeta del sitio web predeterminado.

Por ejemplo, para hacer una prueba, podríamos colocar un archivo llamado "hola.asp" en la carpeta `C:\inetpub\wwwroot`. Para acceder a este archivo desde nuestro explorador deberíamos escribir la dirección `http://localhost/hola.asp`

Si deseamos un código simple para el archivo `hola.asp`, en el que se pruebe si está funcionando o no el motor ASP, podemos utilizar el siguiente:

```

<html>
<head>
<title>Pobando ASP</title>
</head>
<body>
<%
for i=1 to 7
response.write "<font size=" & i & ">Hola Mundo!</font><br>"
next
%>
</body>
</html>

```

Atención: No se debe acceder al archivo utilizando una ruta como esta:
C:\Inetpub\wwwroot\hola.asp, pues de esa manera no se estaría pasando a través del servidor web y la página ASP no se ejecutaría.

Si tuviéramos un sitio medianamente complejo, convendría crear una carpeta dentro de wwwroot con el nombre de nuestro sitio y dentro podríamos colocar todos los archivos. Si el directorio tuviera una ruta como C:\Inetpub\wwwroot\mi_web, accederíamos colocando esta dirección en nuestro navegador: http://localhost/mi_web.

Cuando se accede a ese directorio se sirve el documento por defecto que se haya configurado en el servidor web. El documento por defecto es aquel que se sirve si no se ha especificado ninguno en la ruta de acceso, es decir, si no indicamos ningún archivo en la dirección que colocamos en Internet Explorer (una dirección como http://localhost/mi_web especifica un directorio, pero ningún archivo) se sirve el documento por defecto.

Referencia: Se habla más sobre el documento por defecto en un artículo del manual de publicar en Internet. <http://www.desarrolloweb.com/articulos/200.php?manual=3>

En principio, el documento por defecto en IIS está configurado a Default.asp o Default.htm. Esto quiere decir que deberíamos colocar un archivo con uno de esos nombres en nuestro directorio para que se sirva si el usuario no indica ningún nombre de archivo. Luego veremos cómo cambiar esta configuración.

Para probar todo esto, podemos crear un archivo llamado Default.asp y guardarlo en nuestro directorio C:\Inetpub\wwwroot\mi_web. El código que podríamos utilizar sería por ejemplo este:

```

<html>
<head>
<title>Archivo por defecto en mi_web</title>
</head>
<body>
<h1>Archivo por defecto en mi_web</h1>
<%
for i=0 to 9
response.write i
next
%>
</body>
</html>

```

Recordar que para ver este archivo habría que componer una dirección como esta http://localhost/mi_web. Automáticamente se sirve el archivo Default.asp, aunque no se especifique nada, pues ese es el documento por defecto. También podríamos acceder al archivo especificando su ruta completa: http://localhost/mi_web/Default.asp

Nota: Podemos llamar al archivo Default.asp o bien default.asp (con mayúscula o minúscula en su inicial). Cualquiera de las dos opciones es válida, pues en Windows no se tienen en cuenta las mayúsculas y minúsculas en nombres de archivos.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Administración de IIS

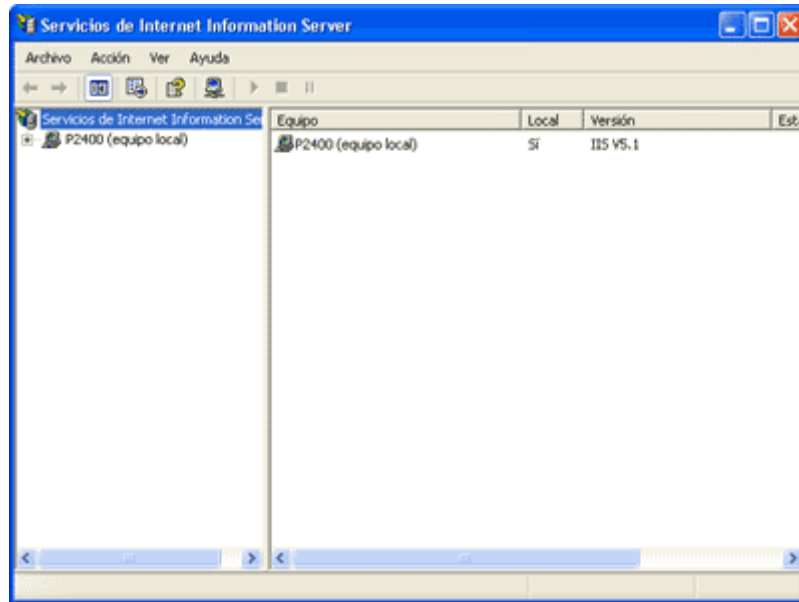
Para administrar el servidor Internet Information Server en Windows XP, disponemos de un panel de control llamado "Servicios de Internet Information Server" al que podemos acceder de varias maneras.

1) Pulsando con el botón derecho en MI PC y seleccionando la opción que pone "Administrar". Esto nos abre "Microsoft Management Console" o, dicho en castellano, la "Administración de equipos". En la lista de la izquierda, en la parte de abajo aparece "Servicios y aplicaciones", entre los que encontraremos una opción buscada: "Servicios de Internet Information Server"

2) Podemos acceder desde el panel de control. Si tenemos configurada la vista clásica encontraremos un icono que pone "Herramientas administrativas" y haciendo doble clic, encontraremos el icono para administrar IIS. Si teníamos configurada la vista por categorías del panel de control (la que aparece por defecto en Windows XP) la búsqueda de la opción es un poco más compleja: Seleccionamos "Rendimiento y mantenimiento" y dentro ya encontraremos el icono de "Herramientas administrativas", al que tenemos que hacer doble clic para encontrar, entre otros, el icono para acceder a "Servicios de Internet Information Server".

3) Otra manera de acceder aparece en la ayuda de Internet Information Server. Se trata de hacer una búsqueda del archivo llamado "inetmgr.exe". Una vez localizado se puede ejecutar y aparece la consola de administración de IIS. Si se desea, se puede hacer un acceso directo a dicho archivo para no tener que buscarlo cada vez que se desee ejecutar.

Una vez hemos accedido al panel "Servicios de Internet Information Server" tenemos ante nosotros la posibilidad de configurar nuestro servidor web en muchos aspectos, por ejemplo podemos, definir el documento por defecto, crear directorios virtuales, modificar las opciones de seguridad, etc.



Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

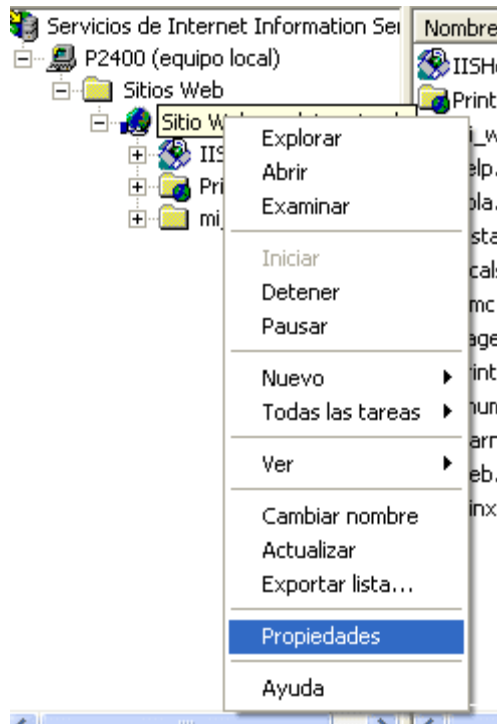
Definir el documento por defecto en IIS

Hablamos antes sobre el documento por defecto, que en IIS viene definido en un principio en los archivos default.asp, default.htm o index.htm. Estos archivos son muy específicos de Windows, pero no suelen utilizarse en la mayoría de los proveedores, así que es mejor que utilicemos un documento por defecto idéntico al que utilizan la mayoría de los proveedores de hosting.

Si nuestro documento por defecto no es el mismo en local (nuestro servidor IIS) y en remoto (espacio en nuestro albergue en un servidor web conectado permanentemente a Internet), puede que tengamos que realizar algunas modificaciones en los nombres de los archivos, para que todo funcione perfectamente al subir el sitio web a Internet, lo que resultaría un engorro adicional e innecesario, de haber configurado nuestro servidor desde un principio.

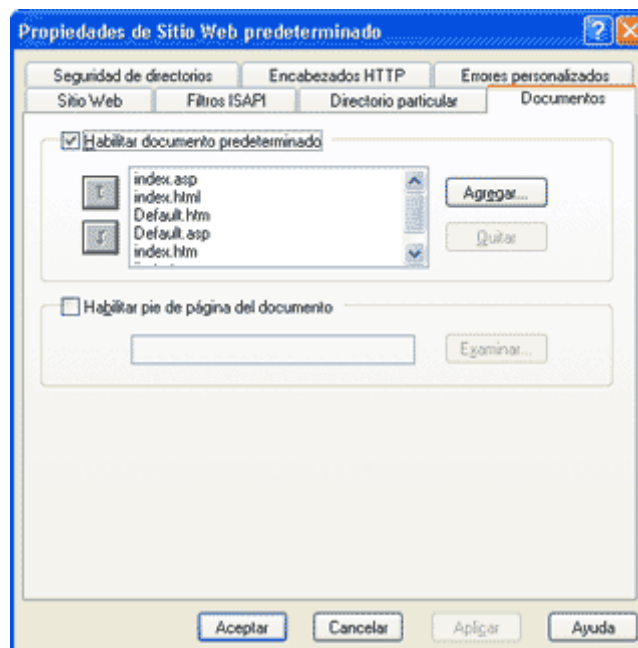
Documentos por defecto típicos son index.html, index.asp (Si es que estamos programando páginas ASP) o index.php (si es que estamos programando con PHP).

Para definir el documento por defecto debemos apretar con el botón derecho el sitio web que deseamos modificar y seleccionar la opción "Propiedades".



Entonces aparece la ventana de propiedades de este sitio, donde debemos seleccionar la pestaña marcada como "Documentos" para poder definir el documento o los documentos por defecto.

Podemos definir uno o varios documentos por defecto, de modo que, si no existe el primer archivo seleccionado como documento por defecto, se intentaría con el segundo, el tercero y cuantos haya configurados hasta que se encuentre un archivo que mostrar o se acabe la lista. Por tanto, el orden de los distintos archivos configurados como documento por defecto si importa y se puede modificar utilizando las flechas de la izquierda de la lista de posibles documentos.



Si no hay ningún archivo en el directorio cuyo nombre sea alguno de los documentos por defecto, no se mostraría ningún archivo y en su lugar recibiríamos un error 404 o el listado de ese directorio, depende de cómo esté configurado IIS para este caso.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Directorios virtuales en IIS

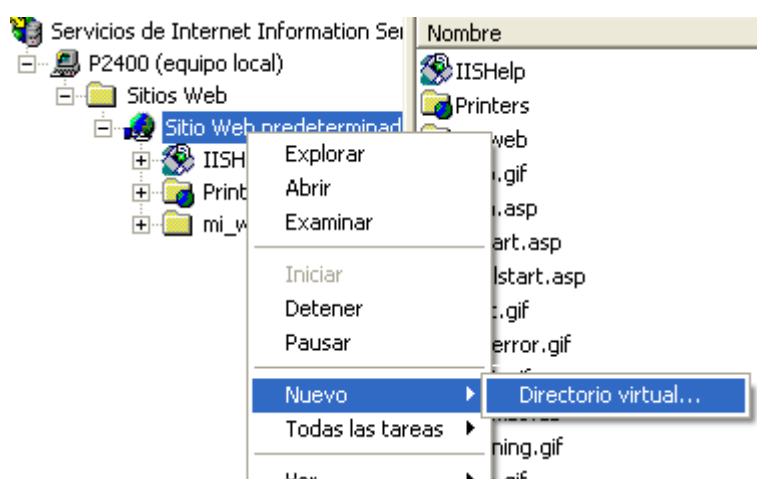
Un directorio virtual es un directorio del servidor que no está dentro del directorio de publicación habitual, es decir, un directorio que no depende de C:\Inetpub\wwwroot pero que sí que se puede acceder a través del servidor web como si estuviera dentro de dicho directorio.

Como ya habíamos comentado, para acceder a nuestro IIS necesitamos escribir una dirección como esta: http://localhost. Así se accede al directorio C:\Inetpub\wwwroot, que es llamado directorio particular. El directorio virtual se accede con algo como http://localhost/directorio_virtual, pero no tiene porque existir una correspondencia en disco de este directorio dentro de la carpeta de publicación, es decir, no tiene porque existir el directorio C:\Inetpub\wwwroot\directorio_virtual, sino que dicho directorio podría estar en cualquier otro sitio de nuestro disco duro, por ejemplo C:\mis_paginas.

Los directorios virtuales se pueden mapear hacia otro directorio de nuestros discos duros o incluso a otro directorio situado en otro ordenador de la red.

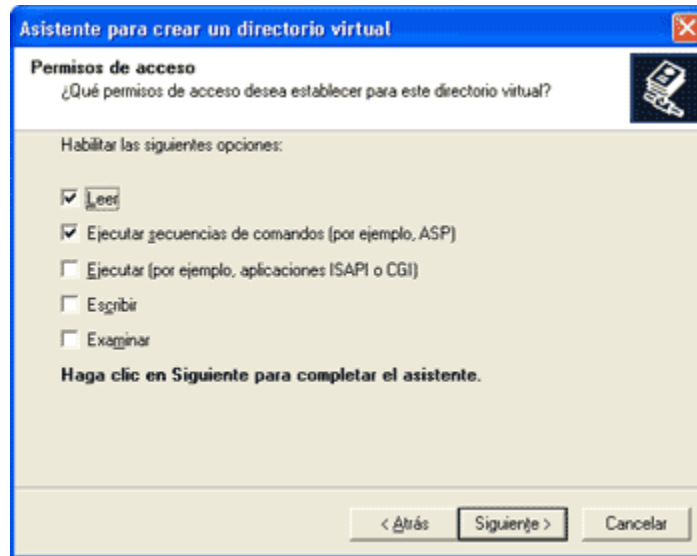
Crear un directorio virtual

Para definir un directorio virtual se puede pulsar con el botón derecho del ratón sobre el sitio web en el que queremos definirlo y seleccionar la opción "Nuevo > Directorio Virtual...". Entonces aparece un asistente que nos guiará paso a paso en el proceso.



El primer paso del asistente nos pregunta el "alias" o nombre lógico que queremos darle al directorio. El segundo paso nos pide la localización física de ese directorio en nuestro disco duro o en la red local. Finalmente nos solicita los permisos que deseamos asignar a ese directorio. El permiso de lectura y el de ejecución de

secuencias de comandos (Por ejemplo, ASP) suelen ser suficientes para la mayoría de los casos.



Una vez finalizado el asistente queda creado el directorio virtual y podremos accederlo a través del alias que hayamos seleccionado.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Envío de formularios por mail

Una de las funcionalidades importantes que tiene que implementar una página bien acabada es el contacto con el administrador del sitio, de modo que los visitantes puedan enviar sus comentarios, ruegos o preguntas.

Existen muchas opciones para implementar un sistema donde los usuarios y el administrador puedan intercambiar mensajes. En este sentido, ya ofrecimos en DesarrolloWeb un artículo que comentaba algunos de ellos: [Contacto con el navegante](#), donde ofrecimos varias ideas muy simples de implementar en páginas sin muchos recursos técnicos. Ahora vamos a ampliar esta información para usuarios que deseen profundizar en este asunto y mejorar el sistema de envío de sus formularios.

Uno de los métodos de contacto con el navegante, tal vez el más habitual, se trata de crear un formulario que el visitante puede rellenar con su información personal y los comentarios al administrador y posteriormente enviar por correo electrónico. En el artículo de [Contacto con el navegante](#) explicamos la manera de construir dicho formulario y configurarlo para el envío de su contenido en un correo electrónico. El envío del correo con el formulario se efectuaba a través del programa de correo que tuviera instalado el visitante en su ordenador, por ejemplo Outlook Express. El problema en este caso es que, si el ordenador del usuario no tiene programa de correo o éste se encuentra mal configurado, el formulario no se podrá enviar.

Otros inconvenientes de mandar el formulario utilizando el cliente de correo electrónico, consisten en que no se puede ofrecer una página que confirme el envío del correo y agradezca a los visitantes el tiempo utilizado en rellenar el formulario. Todo esto resulta poco interesante, molesto y engorroso.

En este artículo vamos a relatar, más concretamente, las distintas opciones de implementar un formulario que sea enviado automáticamente por el servidor de la página después de su rellenado.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Tecnologías para enviar formularios por el servidor

Siempre que sea posible, hay que utilizar el servidor donde está alojada la página para mandar el correo de los formularios. El servidor deberá permitir el envío de correos y estar bien configurado para ello.

Comprobando esto, estaremos seguros que los correos se podrán enviar sin problemas para todos los visitantes, ya que la información no se manda a través del ordenador del usuario -que no sabíamos si se encontraba bien configurado- sino a través del servidor -que hemos comprobado convenientemente-.

Distintas tecnologías

El envío de correos por el servidor se tiene que realizar mediante [programación en el servidor](#). Este tipo de programación es aquel en el que el procesamiento se realiza en el servidor y el recurso disponible para realizar acciones es el propio servidor.

Referencia: El que no sepa qué tipo de programación es la del servidor y desee aclarar ese concepto, debería leerse el [manual de Páginas dinámicas](#) o el de [Introducción a los lenguajes del web](#).

En este artículo vamos a señalar tres tecnologías con las que implementar el envío de correos y alguna que otra indicación sobre cómo realizarlo. Las tres tecnologías son CGI, ASP y PHP. Según nuestro caso deberemos elegir una de ellas, la que más nos convenga, o bien, la que tengamos a nuestra disposición en el lugar de alojamiento de nuestras páginas.

Si estamos trabajando con nuestro propio servidor no será difícil saber cuál de estas tecnologías tenemos disponible, porque seguramente ya estemos utilizando alguna de ellas. Por ejemplo, si hemos instalado [Personal Web Server](#) o [IIS](#), tendremos disponible la programación en [ASP](#). Si tenemos un [Linux](#) con [Apache](#) podríamos utilizar [PHP](#) o [CGI](#).

Referencia: Todo lo necesario para empezar a programar en ASP y PHP lo podemos encontrar en nuestras secciones:
[ASP a fondo](#)
[PHP a fondo](#)

Si tenemos la página colgada de un proveedor de hosting deberíamos preguntar al servicio técnico del proveedor cuál de estas opciones está disponible para realizar el envío de formularios por el servidor. La mayoría de los proveedores tienen ya instalados los recursos con los que realizar esta tarea, ya que es una de las que necesitan los desarrolladores más habitualmente. En algunos casos os darán varias opciones e incluso tendrán a vuestra disposición tutoriales donde explican el uso de las herramientas disponibles.

También podríamos hacer este ejemplo con programación en [Java](#) -Servlets o [Java Server Pages](#)- o con otros lenguajes como Cold Fusion. Sin embargo, en este artículo vamos a centrarnos en las tecnologías más habituales, o más populares, en el desarrollo de webs.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Envío de formularios por mail usando CGI

El método de envío más sencillo es a través de un programa [CGI](#). CGI es la tecnología para programación del servidor más antigua, tal vez un poco obsoleta en cuanto a metodología puesto que no ha evolucionado mucho en los últimos años.

También resulta un poco más complicado de aprender que las otras opciones señaladas en este artículo, como ASP o PHP. Por todo ello, empezar desde cero a crear un sistema CGI resulta, en nuestra opinión, desaconsejado. En lugar de construir nuestro propio CGI nos parece más interesante utilizar alguno de los ya creados.

CGI FormMail

Es uno de los programas CGI que se puede utilizar para el envío de formularios. Su dirección www.scriptarchive.com/formmail.html permite la descarga de los archivos y documentación para configurarlo y crear los formularios que utilicen el CGI para enviarse por correo.

Otros sitios con programas CGI

Existen sitios donde podemos encontrar un directorio de programas CGI clasificados por categorías. En nuestro caso podríamos echar un vistazo a la sección de Formularios o Envío de correo para encontrar otras opciones y examinar la oferta disponible para realizar estas tareas.

Podemos acceder a nuestro [directorio de recursos para CGI](#), donde podremos encontrar algunos enlaces interesantes.

Es importante señalar que, si estamos alojando nuestras páginas en algún proveedor de alojamiento, éstos seguramente tengan instalado y configurado algún programa de envío de formularios, por lo que lo mejor sería preguntar a los técnicos de nuestro servicio de hosting cuál es y la manera de funcionar que tiene. Generalmente todos los proveedores de pago tienen algún tipo de CGI preinstalado y, aunque algunos alojadores gratuitos también disponen de alguno, no les podemos exigir que así sea.

Ejemplo de código de utilización de FormMail

No pretendemos explicar el manejo de un CGI para enviar correos, puesto que lo mejor sería que aprendiésemos a utilizar el CGI que tengamos disponible en cada caso en nuestro alojamiento.

De todos modos, incluimos aquí un código de un formulario que se enviaría por email utilizando el CGI FormMail. Podemos ver como el atributo action del formulario está dirigido hacia el CGI que se encargará de enviarlo por mail. Asimismo, podemos ver una serie de campos hidden que tienen como objetivo configurar el correo a enviar y señalar la dirección a la que redirigir el navegador después de haber enviado el correo, que simplemente tendría un texto dando las gracias al visitante.

```
<FORM ACTION="http://cgi.dominioxyz.com/FormMail.pl" METHOD="POST">
<input type=hidden name="recipient" value="pepe@pepe.com">
<input type=hidden name="subject" value="Asunto del mail que se envia">
<input type=hidden name="redirect" value="http://www.dominioxyz.com/gracias.html">

<table border="0" cellspacing="2" cellpadding="2">
  <tr>
    <td><font face="Verdana, Arial, Helvetica, sans-serif" size="2">Nombre:<br>
      <input type="text" name="nombre" size="25" maxlength="300">
    </font></td>
    <td><font face="Verdana, Arial, Helvetica, sans-serif" size="2">Email:<br>
      <input type="text" name="email" size="25" maxlength="300">
    </font></td>
  </tr>
  <tr>
    <td><font size="2">Dirección:<br>
      <input type="text" name="direccion" size="30" maxlength="500">
    </font></td>
    <td><font size="2">Telefono:<br>
      <input type="text" name="telefono" size="12" maxlength="15">
    </font></td>
  </tr>
  <tr>
    <td colspan="2"><font face="Verdana, Arial, Helvetica, sans-serif" size="2">Observaciones:<br>
      <textarea name="observaciones" cols="50" rows="5"></textarea>
    </font></td>
  </tr>
  <tr>
    <td colspan="2"></td>
  </tr>
</table>
```

```

        <div align="center"><br>
        <input type="submit" value="Enviar">
    </td>
</tr>
</table>
</form>

```

Informe de **Miguel Angel Alvarez**
 Director desarrolloweb.com
 Mail: eugim@desarrolloweb.com

Enviar un formulario por email con ASP

Si deseamos que, al pulsar el botón de envío de un formulario, se manden los datos por email utilizando el servidor y sin estar supeditados a la configuración del cliente para saber si ciertamente ese mensaje se pudo enviar, debemos utilizar alguna tecnología de programación de páginas del lado del servidor. En este artículo vamos a presentar un ejemplo sobre cómo hacer esta tarea con ASP.

ASP (Active Server Pages) es la tecnología de scripting del lado del servidor desarrollada por Microsoft. Con este tipo de programación podemos construir páginas que muestren un formulario y manden un correo electrónico automáticamente con los datos recibidos, una vez enviado.

Podríamos utilizar un esquema de código como el siguiente:

```

<%
if request.form="" then
'no recibo formulario, entonces lo muestro
%>
<form action="formulario_mail_asp.asp" method="POST">
Nombre: <input type="Text" name="nombre" size="12" maxlength="200">
<br>
Email: <input type="Text" name="email" size="12" maxlength="200">
<br>
<input type="submit" value="Enviar">
</form> <%
else
'si que recibo un formulario, entonces lo trato
'recojo los datos
nombre = request.form("nombre")
email = request.form("email")
'compongo el cuerpo del mensaje
cuerpo = "Formulario recibido" & VBNEWLINE & VBNEWLINE
cuerpo = cuerpo & "Nombre: " & nombre & VBNEWLINE
cuerpo = cuerpo & "Email: " & email
'mando el correo...
'.....
response.write "Gracias por rellenar el formulario. Se ha enviado correctamente." end if %>

```

En el anterior script utilizamos `if (request.form="")` para saber si estamos recibiendo o no información de un formulario.

Si no hemos recibido nada (porque en `request.form` tenemos una cadena vacía) este `if` saldría por su caso verdadero, en el que tendremos que presentar el formulario en la página.

El caso `else`, cuando sí que recibíamos un formulario, recogemos sus datos y creamos el cuerpo del mail que enviaremos a la dirección del administrador.

El resto del código, utilizado para definir las propiedades del email y enviarlo,

todavía no lo hemos indicado. Antes una aclaración.

ASP no tiene entre las funciones del lenguaje una que sirva para enviar correos electrónicos. Sin embargo, podemos utilizar un [componente ActiveX del servidor](#) para realizar esas acciones.

El [componente CDONTS](#) (presente en muchas de las [instalaciones de IIS](#)) serviría para realizar el envío del mail, pero también existen en el mercado otros componentes de servidor comerciales para realizar esas acciones con mayores funcionalidades. Uno de ellos es [AspEmail](#), que es el que vamos a utilizar en este script, pero no es el único. Hablamos de las distintas opciones en el artículo [Enviar mails desde ASP](#).

El código para enviar un mail podría ser algo como esto:

```
'creo el objeto correo
set mail = server.createObject("Persits.MailSender")
'configuro el mensaje
'señalo el servidor de salida para enviar el correo
mail.host = "mail.tudominio.com"
'indico la dirección de correo del remitente
mail.from = "loquesea@tudominio.com"
'indico la dirección del destinatario del mensaje
mail.addAddress "loquesea@tudominio.com"
'indico el cuerpo del mensaje
mail.body = cuerpo
'lo envío
'aseguro que no se presenten errores en la página si se producen
On Error Resume Next
mail.send
if Err ><0 then
response.write "Error, no se ha podido completar la operación"
else
response.write "Gracias por rellenar el formulario. Se ha enviado correctamente." end if
```

Nos podemos guiar por los comentarios para hacernos una idea del funcionamiento de este trozo de código. Básicamente se crea el objeto mail, que contendrá el correo que se va a generar, y se definen sus propiedades como el servidor de correo a utilizar, el destinatario o el asunto. Luego se envía y se realiza una sencilla tarea de detección de errores para saber si se pudo realizar la tarea con éxito o no y mostrar un mensaje adecuado para cada caso.

Nota: no se pretenden explicar los pormenores de este Control Activex del servidor, ya que hay una excelente documentación y ejemplos en la página web de la empresa que lo ha desarrollado. En caso de que se desee aprender algo más, por favor, dirigiros a la página www.aspemail.com o la del componente del que dispongáis.que dispongáis.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Enviar un formulario por mail con PHP

El método para enviar un formulario con PHP resulta muy similar al [utilizado en ASP](#). Varía tan sólo la sintaxis utilizada y las líneas de código que realizan el envío del correo electrónico.

Para empezar, sería muy útil que [aprendamos a enviar correos electrónicos con PHP](#), para lo que tenemos un artículo en DesarrolloWeb.

Esquema de funcionamiento

En este caso nos vamos a apoyar en la variable `$HTTP_POST_VARS`, que debería contener el formulario, para saber si hemos recibido o no datos desde un formulario. Dicha variable la utilizamos en un enunciado `if (! $HTTP_POST_VARS)`, que si pasa por el caso positivo -no había nada en `$HTTP_POST_VARS`- significa que no se ha recibido nada desde un formulario. En ese caso, muestro el formulario de contacto.

En caso contrario -sí que había algo en `$HTTP_POST_VARS`- quiere decir que estamos recibiendo datos por un formulario y en ese caso, recogemos los datos y componemos el cuerpo del mensaje.

Veamos el código de la página para crear el formulario, recibirlo y componer el cuerpo del mensaje que se va a enviar. En el mismo código podremos encontrar también la llamada a la función que envía el correo.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
  <title>Mádanos tus comentarios</title>
</head>

<body bgcolor="#cccc66" text="#003300" link="#006060" vlink="#006060">
<?
if (!$HTTP_POST_VARS){
?>
<form action="envia_form_php.php" method=post>
Nombre: <input type=text name="nombre" size=16>
<br>
Email: <input type=text name=email size=16>
<br>
Comentarios: <textarea name=coment cols=32 rows=6></textarea>
<br>
<input type=submit value="Enviar">
</form>
<?
}else{
  //Estoy recibiendo el formulario, compongo el cuerpo
  $cuerpo = "Formulario enviado\n";
  $cuerpo .= "Nombre: " . $HTTP_POST_VARS["nombre"] . "\n";
  $cuerpo .= "Email: " . $HTTP_POST_VARS["email"] . "\n";
  $cuerpo .= "Comentarios: " . $HTTP_POST_VARS["coment"] . "\n";

  //mando el correo...
  mail("admin@tudominio.com","Formulario recibido",$cuerpo);

  //doy las gracias por el envío
  echo "Gracias por rellenar el formulario. Se ha enviado correctamente.";
}
?>
</body>
</html>
```

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

PÁGINAS DINÁMICAS

Concepto de páginas dinámicas

Muy probablemente, estimado lector, seas una persona más o menos familiarizada con el lenguaje HTML y con toda seguridad hayas desarrollado algún sitio basado en esta herramienta. En realidad el HTML no es lenguaje de programación sino, más bien, se trata de un lenguaje descriptivo que tiene como objeto dar formato al texto y las imágenes que pretendemos visualizar en el navegador.

A partir de este lenguaje somos capaces de introducir enlaces, seleccionar el tamaño de las fonts o intercalar imágenes, todo esto de una manera prefijada y en ningún caso inteligente. En efecto, el HTML no permite el realizar un simple cálculo matemático o crear una página de la nada a partir de una base de datos. A decir verdad, el HTML, aunque muy útil a pequeña escala, resulta bastante limitado a la hora de concebir grandes sitios o portales.

Es esta deficiencia del HTML la que ha hecho necesario el empleo de otros lenguajes accesorios mucho más versátiles y de un aprendizaje relativamente más complicado, capaces de responder de manera inteligente a las demandas del navegador y que permiten la automatización de determinadas tareas tediosas e irremediables como pueden ser las actualizaciones, el tratamiento de pedidos de una tienda virtual...

Estos lenguajes capaces de recrear a partir de ciertos "scripts" un sinnúmero de páginas automatizadas son los protagonistas de este concepto de páginas dinámicas.

Este manual, que no es más que una introducción a otros manuales en curso de redacción, está destinado a aquellos que sienten que el HTML se les queda corto para realizar sus proyectos y que, sin tener ni idea de programar, desean dar el paso y darle un nuevo aire a sus páginas sin por ello pasar por experiencias traumáticas debidas a su poca afinidad con los lenguajes informáticos.

El contenido ha sido, por lo tanto, deliberadamente simplificado y será ampliado posteriormente a partir de anexos y artículos de mayor nivel para que pueda ser también utilizado por aquellos que ya están familiarizados con este tipo de lenguajes.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Páginas dinámicas vs HTML

A pesar de que las páginas dinámicas nos puedan en un principio limitar a causa de su mayor complejidad con respecto al HTML, todas las ventajas que nos ofrecen compensan con creces este esfuerzo inicial.

No obstante, hay que ser consciente del posible interés que pueda tener para uno el lanzarse en esta aventura de aprender un nuevo lenguaje y volver a rediseñar su propio sitio.

Si la página en la que estamos pensando o que queremos rediseñar es

relativamente pequeña, no necesita estar al día continuamente sino que sus contenidos son perennes y no hemos previsto el pagar por mantenerla, el empleo de páginas dinámicas puede quedarse grande y resultar a todas luces improductivo.

Por el contrario, si el sitio es extenso y sus contenidos cambian rápidamente, nos interesa el automatizar en la medida de lo posible todas las tareas de tal forma que podamos gestionar su explotación de la manera más óptima.

Para dejar más claro hasta que punto resulta útil utilizar páginas dinámicas lo mejor será ejemplificarlo a partir de un sitio web modelo.

Supongamos que hemos decidido realizar un portal de televisión donde una de las informaciones principales a proveer podría ser la programación semanal. Efectivamente, esta información suele ser dada por las televisiones con meses de antelación y podría ser muy fácilmente almacenada en una base de datos. Si trabajásemos con páginas HTML, tendríamos que construir una página independiente para cada semana en la cual introduciríamos "a mano" cada uno de los programas de cada una de las cadenas. Asimismo, cada semana nos tendríamos que acordar de descolgar la página de la semana pasada y colgar la de la anterior. Todo esto podría ser fácilmente resuelto mediante páginas dinámicas. En este caso, lo que haríamos sería crear un programa (solo uno) que se encargaría de recoger de la base de datos de la programación aquellos programas que son retransmitidos en las fechas que nos interesan y de confeccionar una página donde aparecerían ordenados por cadena y por hora de retransmisión. De este modo, podemos automatizar un proceso y desentendernos de un aspecto de la página por unos meses.

Este hecho lo podríamos aplicar a otras situaciones: podemos preparar el horóscopo de todos los días, las promociones de un sitio de e-comercio...

Además, tampoco resultaría complicado el introducir una pequeña caja de búsqueda que nos permitiera dar rápidamente con el programa que queremos ver, saber a qué hora y en qué cadena se emite.

Volviendo a nuestro portal de televisión, en él hay una sección en la cual presentamos todas las series actualmente emitidas con comentarios sobre ella, fotos, etc. Podríamos, en lugar de hacer una página HTML por serie, hacer una única página dinámica en contacto con una base de datos en la cual visualizamos las fotos y comentarios relativos a la serie que nos interesa. Asimismo, si lo que buscamos es modificar el formato del texto de dicha sección, podemos automatizar este proceso sin necesidad de cambiar a mano cada una de las etiquetas font y sin hacer uso de la hojas de estilo las cuales no son reconocidas por la totalidad de los navegadores.

Otra serie de aspectos tales como la gestión de las lenguas, podrían ser fácilmente resueltos sin para ello duplicar el número de páginas y buscar los textos a traducir penosamente entre el código HTML.

En realidad, a partir de estas herramientas, podemos plantearnos cuantas cosas queramos. El único límite... nuestra imaginación

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Lenguajes de lado servidor o cliente

El navegador es una especie de aplicación capaz de interpretar las órdenes recibidas en forma de código HTML fundamentalmente y convertirlas en las páginas que son el resultado de dicha orden.

Cuando nosotros pinchamos sobre un enlace hipertexto, en realidad lo que pasa es que establecemos una petición de un archivo HTML residente en el servidor (un ordenador que se encuentra continuamente conectado a la red) el cual es enviado e interpretado por nuestro navegador (el cliente).

Sin embargo, si la página que pedimos no es un archivo HTML, el navegador es incapaz de interpretarla y lo único que es capaz de hacer es salvarla en forma de archivo. Es por ello que, si queremos emplear lenguajes accesorios para realizar un sitio web, es absolutamente necesario que sea el propio servidor quien los ejecute e interprete para luego enviarlos al cliente (navegador) en forma de archivo HTML totalmente legible por él.

De modo que, cuando pinchamos sobre un enlace a una página que contiene un script en un lenguaje comprensible únicamente por el servidor, lo que ocurre en realidad es que dicho script es ejecutado por el servidor y el resultado de esa ejecución da lugar a la generación de un archivo HTML que es enviado al cliente.

Así pues, podemos hablar de lenguajes de lado servidor que son aquellos lenguajes que son reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente en un formato comprensible para él. Por otro lado, los lenguajes de lado cliente (entre los cuales no sólo se encuentra el HTML sino también el Java y el JavaScript los cuales son simplemente incluidos en el código HTML) son aquellos que pueden ser directamente "digeridos" por el navegador y no necesitan un pretratamiento.

Cada uno de estos tipos tiene por supuesto sus ventajas y sus inconvenientes. Así, por ejemplo, un lenguaje de lado cliente es totalmente independiente del servidor, lo cual permite que la página pueda ser albergada en cualquier sitio sin necesidad de pagar más ya que, por regla general, los servidores que aceptan páginas con scripts de lado servidor son en su mayoría de pago o sus prestaciones son muy limitadas. Inversamente, un lenguaje de lado servidor es independiente del cliente por lo que es mucho menos rígido respecto al cambio de un navegador a otro o respecto a las versiones del mismo. Por otra parte, los scripts son almacenados en el servidor quien los ejecuta y traduce a HTML por lo que permanecen ocultos para el cliente. Este hecho puede resultar a todas luces una forma legítima de proteger el trabajo intelectual realizado.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Lenguajes de lado servidor

Existe una multitud de lenguajes concebidos o no para Internet. Cada uno de ellos explota más a fondo ciertas características que lo hacen más o menos útiles para desarrollar distintas aplicaciones.

La versatilidad de un lenguaje está íntimamente relacionada con su complejidad. Un lenguaje complicado en su aprendizaje permite en general el realizar un espectro

de tareas más amplio y más profundamente. Es por ello que a la hora de elegir el lenguaje que queremos utilizar tenemos que saber claramente qué es lo que queremos hacer y si el lenguaje en cuestión nos lo permite o no.

En el dominio de la red, los lenguajes de lado servidor más ampliamente utilizados para el desarrollo de páginas dinámicas son el ASP, PHP y PERL.

El ASP (Active Server Pages) es un lenguaje derivado del Visual Basic desarrollado por Microsoft. Evidentemente su empleo se realiza sobre plataformas funcionando bajo sistema Windows NT.

El PHP podría ser considerado como el lenguaje análogo al ASP utilizado en plataformas Unix y Linux.

Estos dos lenguajes resultan bastante útiles para la explotación de bases de datos y su aprendizaje resulta accesible para una persona profana de la programación. Cualquiera de ellos resultaría la opción ideal a la hora de hacer evolucionar un sitio web realizado en HTML.

Por otra parte, el PERL es un lenguaje más rápido y potente que requiere obviamente un aprendizaje más largo y resulta más reservado para personas ya familiarizadas con la verdadera programación.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Algunos aspectos prácticos previos

Antes de lanzarnos en las consideraciones teóricas relativas a la programación, resultaría interesante aclarar algunas dudas que puede presentarse referentes a cómo escribir y publicar páginas dinámicas.

Para escribir una página dinámica podemos hacerlo del mismo modo que si lo hiciésemos en HTML. En realidad, el código está constituido exclusivamente de texto y lo único que tenemos que hacer por lo tanto es guardar el archivo texto con una extensión que pueda ser reconocida posteriormente por el servidor. Así, por ejemplo, las páginas de ASP son reconocidas por su extensión "asp" del mismo modo que las de PHP lo son a partir de extensiones "php" u otras en las que se especifica la versión utilizada ("php3" o "php4"). En muchos casos el servidor nos permite seleccionar qué tipo de extensión debe ser reconocida para un determinado lenguaje por lo que estas extensiones no están totalmente generalizadas aunque son sin duda las más utilizadas.

Dado que se trata únicamente de archivos texto, es posible crear páginas dinámicas a partir del [Bloc de Notas](#) o cualquier otro procesador de texto plano (Texto ASCII, sin códigos raros como los que pone MS Word). También podemos utilizar los editores clásicos empleados para el HTML aunque en este caso, estamos obligados a trabajar en modo editar y no en modo gráfico. Esta última posibilidad resulta tanto menos aconsejable cuanto que la mayoría de estos editores no están preparados para la programación en estos lenguajes y algunos de ellos ([Frontpage](#) en sus versiones anteriores a la 2000, sin ir más lejos) están dispuestos a borrar aquellos textos que no es capaz de interpretar.

Existe un artículo en [desarrolloweb](#) que explica de manera más detallada el mundo de los [editores de páginas](#)

Existen sin embargo algunos editores de HTML que si ofrecen ventajas al editar scripts. Tal es el caso del Homesite que muestra coloraciones diferentes en función de la sintaxis del programa lo cual permite una lectura más fácil. Además, hay otra serie de editores más pensados para páginas dinámicas en general o para algún lenguaje en particular.

Una vez el programa realizado, el paso inmediato es el de ejecutarlo. Como ya ha sido explicado, los lenguajes de lado servidor ejecutan los scripts en el propio servidor y envían el resultado en forma de código HTML al cliente (navegador). Resulta obvio que para probar entonces el programa es necesario colgar por FTP los archivos que lo componen en el servidor y hacer la petición desde el navegador. En principio, no es por tanto posible el trabajar offline a partir de archivos alojados en el disco duro tal y como hacíamos con el HTML. Esto en realidad no es completamente cierto ya que existe la posibilidad de convertir nuestro propio ordenador en servidor web personal de manera que podemos trabajar en local sin necesidad de estar conectados continuamente lo cual podría representar un problema para aquellos que tengan que pagar una factura telefónica al estar conectados por modem además de resultar más juicioso puesto que un servidor no es el sitio ideal para hacer nuestros pinitos en un lenguaje que no controlamos suficientemente. Bucles infinitos, variables no cerradas y otra serie de irregularidades pueden estar consumiendo recursos importantes en perjuicio de los usuarios que estén accediendo a otras paginas albergadas por este servidor.

Como puede verse, la forma de operar resulta casi análoga a lo que hacíamos para nuestro sitio estático y no presenta ninguna complicación aparente. Cabe destacar que, como ya se ha dicho anteriormente, para poder servir de estos lenguajes de lado servidor, es imprescindible que el servidor esté preparado para leer las páginas programadas en un lenguaje no comprendido por el navegador. Dichos servidores son en su gran mayoría de pago lo cual añade ciertas limitaciones económicas al proyecto.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Conceptos básicos de programación I

Antes de abordar en detalle las particularidades de estos lenguajes, es importante guardar en espíritu toda una serie de nociones básicas comunes. Estos aspectos son sin duda conocidos por aquellos que hayan programado alguna vez y pueden ser muy rápidamente asimilados por todos los que estén familiarizados con las matemáticas. Teniendo en cuenta esto, hemos querido acercar estos conceptos a cualquier persona proponiendo definiciones poco rigurosas y carentes de detalles pero que en contrapartida permiten ser digeridas con más facilidad.

Así pues, aquellos sugerimos el pasar directamente al siguiente capítulo a todos aquellos que consideren conocer perfectamente los conceptos de variable y función aunque siempre puede resultar interesante volver a recordarlo visto desde el prisma de otra definición.

Variable

Una variable consiste en un elemento al cual le damos un nombre y le atribuimos un determinado tipo de información. Las variables pueden ser consideradas como la base de la programación.

De este modo podríamos escribir en un lenguaje ficticio:

```
a="perro"  
b="muerde"
```

La variable que nosotros llamamos "a" posee un elemento de información de tipo texto que es "perro". Asimismo, la variable "b" contiene el valor "muerde".

Podríamos definir una tercera variable que fuese la suma de estas dos:

```
c=a+b
```

Si introduyésemos una petición de impresión de esta variable en nuestro lenguaje ficticio:

```
imprimir(c)
```

El resultado podría ser:

```
perro muerde
```

Podríamos de la misma forma trabajar con variables que contuviesen números y construir nuestro programa:

```
a=3  
b=4  
c=a+b  
imprimir(c)
```

El resultado de nuestro programa sería:

```
7
```

La utilidad de estas variables quedará de relieve en el transcurso de los siguientes capítulos.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Conceptos básicos de programación II

Funciones y procedimientos

La función podría ser definida como un conjunto de instrucciones que permiten procesar las variables para obtener un resultado. Puede que esta definición resulte un poco vaga si no nos servimos de un ejemplo para ilustrarla.

Supongamos que queremos calcular el valor total de un pedido a partir de la simple suma de los precios de cada uno de los artículos. Podríamos definir una función suma en nuestro lenguaje ficticio:

```
definir funcion suma(art1,art2,art3)  
suma=art1+art2+art3  
imprimir(suma)
```

fin funcion

Este supuesto programa nos permitiría calcular la suma de tres elementos e imprimir el resultado en pantalla. Lo interesante de utilizar este tipo de funciones es que ellas nos permiten su utilización sistemática tantas veces como queramos sin necesidad de escribir las instrucciones tantas veces como veces queremos utilizarla. Por supuesto, podemos prescindir de esta declaración de función e introducir una línea del siguiente tipo:

```
imprimir(art1+art2+art3)
```

Evidentemente, cuanto más complicada sea la función y más a menudo la utilicemos en nuestros scripts más útil resulta definir las.

Esta función suma podría ser utilizada en cualquier lugar de nuestro script haciendo una llamada del siguiente tipo:

```
ejecuta suma(4,6,9)
```

Cuyo resultado sería:

19

Del mismo modo, los procedimientos son parecidos a las funciones. La diferencia consiste tan solo en que en estos últimos el interés no radica en el resultado obtenido sino más bien en las operaciones realizadas al ejecutarla (creación de un archivo, reenvío a otra página,...). En lenguajes como el PHP las funciones y los procedimientos son considerados como la misma cosa y para definirlos se hace usando los mismos comandos.

Tanto las variables como las funciones y los procedimientos deben ser nombradas sin servirse de acentos, espacios ni caracteres especiales para no correr riesgos de error .

Estos conceptos son básicos para una comprensión de la programación. No obstante, es posible que si es la primera vez que oímos hablar de ellos, su asimilación puede resultar parcial o nula. En realidad esto no es preocupante ya que a partir de los ejemplos de los capítulos siguientes y con la práctica de uno mismo se irán consolidando poco a poco.

Para nada hay que desanimarse si después de leer este capítulo algunas dudas quedan en el aire.

El paso siguiente es continuar el aprendizaje de un lenguaje de programación que nos sirva para construir las páginas dinámicas, con el [manual de ASP](#) o el [manual de PHP](#), según nuestros gustos o necesidades.

Informe de **Rubén Alvarez**
Mail: ruben@desarrolloweb.com

Enlaces sin subrayado

Una de las cosas que me preguntan a menudo los usuarios cuando me mandan correo a eugim@desarrolloweb.com es cómo quitarle el subrayado a los enlaces. He aquí la respuesta que le di a uno de nuestros amigos:

Tienes dos maneras de aprender a hacer esto:

- Por la vía rápida, más fácil hoy pero peor para en el futuro.
- Aprendiendo bien el asunto, desde el principio.

RÁPIDO

solo ponle el atributo - **style="text-decoration:none"** - al enlace que quieras sin subrayar. Ejemplo:

```
<a href="http://www.desarrolloweb.com" style="text-decoration:none"> desarrolloweb</a>
```

APRENDIENDO BIEN

Eso se hace con **Hojas de Estilo en Cascada**, también llamadas **CSS**, incluso la forma en la rápida estás utilizándolas, posiblemente sin saberlo. Si aprendes CSS podrás saber cómo quitar el subrayado de manera más eficaz, por ejemplo, quitándole de una sola vez los subrayados a todos los enlaces de la página o del sitio entero. Además aprenderás muchas otras cosas muy importantes y chulas.

Puedes aprender a trabajar con las Hojas de Estilo en la dirección www.desarrolloweb.com/manuales/2

Tu eliges, **pero no se te ocurra hacer un sitio entero**, o incluso una página grande donde todos los enlaces estén sin subrayar, **con el método rápido** porque a la larga es más lento y costoso. El manual que te propongo es corto de leer pero preciso y fácil de entender, te ayudara mucho en tu evolución como diseñador web.

Eso es todo!

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Formatos gráficos para páginas web

El componente gráfico de las páginas web tiene mucha importancia, es el que hace que estas sean vistosas y el que nos permite aplicar nuestra creatividad para hacer del diseño de sitios una tarea agradable. Es también una herramienta para acercar los sitios al mundo donde vivimos, si embargo, es también el causante de errores graves en las páginas y hacer de estas, en algunos casos, un martirio para el visitante.

Las nociones básicas para el uso de archivos gráficos son sencillas, conocerlas, aunque sea ligeramente, nos ayudará a crear sitios agradables y rápidos. No cometer errores en el uso de las imágenes es fundamental, aunque no seas un

diseñador y las imágenes que utilices sean feas, utilízalas bien y así estarás haciendo más agradable la visita a tus páginas.

Tipos de archivos

En Internet se utilizan principalmente dos tipos de archivos gráficos GIF y JPG, pensados especialmente para optimizar el tamaño que ocupan en disco, ya que los archivos pequeños se transmiten más rápidamente por la Red.

El formato de archivo GIF se usa para las imágenes que tengan dibujos, mientras que el formato JPG se usa para las fotografías. Los dos comprimen las imágenes para guardarlas. La forma de comprimir la imagen que utiliza cada formato es lo que los hace ideales para unos u otros propósitos.

Adicionalmente, se puede usar un tercer formato gráfico en las páginas web, el PNG. Este formato no tiene tanta aceptación como el GIF o JPG por varias razones, entre las que destacan el desconocimiento del formato por parte de los desarrolladores, que las herramientas habituales para tratar gráficos (como por ejemplo Photoshop) generalmente no lo soportan y que los navegadores antiguos también tienen problemas para visualizarlas. Sin embargo, el formato se comporta muy bien en cuanto a compresión y calidad del gráfico conseguido, por lo que resultaría útil si se llega a extender su uso.

GIF

A parte de ser un archivo ideal para las imágenes que estén dibujadas tiene muchas otras características que son importantes y útiles.



Un logotipo es un ejemplo claro de imagen GIF

Compresión: Es muy buena para dibujos, como ya hemos dicho. Incluso puede ser interesante si la imagen es muy pequeña, aunque sea una foto.

Transparencia: es una utilidad para definir ciertas partes del dibujo como transparentes. De este modo podemos colocar las imágenes sobre distintos fondos sin que se vea el cuadrado donde está inscrito la el dibujo, viendose en cambio la silueta del dibujo en cuestión.

Para crear un gif transparente debemos utilizar un programa de diseño gráfico, con el podemos indicar qué colores del dibujo queremos que sean transparentes. Generalmente, definimos la transparencia cuando vamos a guardar el gráfico.



Parte de esta imagen es transparente

Colores: Con este formato gráfico podemos utilizar paletas, conjuntos, de 256 colores o menos. Este es un detalle muy importante, puesto que cuantos menos colores utilicemos en la imagen, por lo general, menos ocupará el archivo. En ocasiones, aunque utilicemos menos colores en un gráfico, este no pierde mucho en calidad, llegando a ser inapreciable a la vista.

En algunos programas podemos modificar la cantidad de colores al guardar el archivo, en otros lo hacemos mientras creamos el gráfico.



32 colores



16 colores



8 colores

Imagen tomada con distintas paletas de colores. Se puede apreciar como con pocos colores se ve bien el gráfico y como pierde un poco a medida que le restamos colores.

JPG

Veamos ahora cuales son las características fundamentales del formato JPG:

Compresión: Tal como hemos dicho anteriormente, su algoritmo de compresión hace ideal este formato para guardar fotografías. Además, con JPG podemos definir la calidad de la imagen, con calidad baja el fichero ocupará menos, y viceversa.



Una fotografía con formato JPG

Transparencia: Este formato no tiene posibilidad de crear áreas transparentes. Si deseamos colocar una imagen con un área que parezca transparente procederemos así: con nuestro programa de diseño gráfico haremos que el fondo de la imagen sea el mismo que el de la página donde queremos colocarla. En muchos casos los fondos de la imagen y la página parecerán el mismo.

Colores: JPG trabaja siempre con 16 millones de colores, ideal para fotografías.



Intento de transparencia en JPG. [Pulsar para ampliar](#)

Optimizar ficheros

Para que las imágenes ocupen lo menos posible y se transfieran rápidamente por la Red debemos aprender a optimizar los ficheros gráficos. Para ello debemos hacer lo siguiente:

Para los archivos GIF: Reduciremos el número de colores de nuestra paleta. Esto se hace con nuestro editor gráfico, en muchos casos podremos hacerlo al guardar el archivo.



10,8 KB

GIF 256 colores -



5,5 KB

GIF 32 colores -



KB

GIF 4 colores - 2

Para los archivos JPG: Ajustaremos la calidad del archivo cuando lo estemos guardando. Este formato nos permite bajar mucho la calidad de la imagen sin que esta pierda mucho en su aspecto visual.



JPG
calidad 0
3 KB



JPG
calidad 20
5,9 KB



JPG
calidad 50
10 KB

Es imprescindible disponer para optimizar la imagen de una herramienta buena que nos permita configurar estas características de la imagen con libertad y fácilmente. Photoshop 5.5 o 6 es un programa bastante recomendable, pues incorpora una opción que se llama "Guardar para el Web" con la que podemos definir los colores del gif, calidad del JPG y otras opciones en varias muestras a la vez. Así con todas las opciones configurables, viendo los resultados a la vez que el tamaño del archivo podemos optimizar la imagen de una manera precisa con los resultados que deseamos.

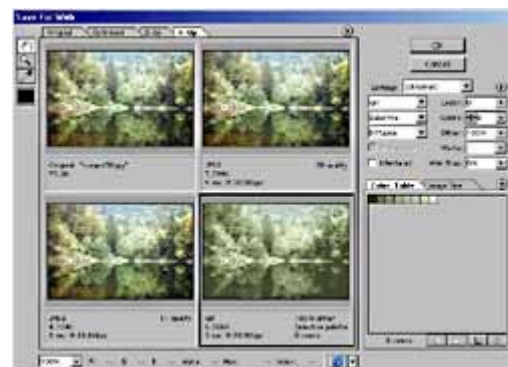
También existen en el mercado otros programas que nos permiten optimizar estas imágenes de manera sorprendente. Una vez hemos creado la imagen la pasamos por estos programas y nos comprimen aun más el archivo, haciéndolo rápido de transferir y, por tanto, más óptimo para Internet. Al ser estas utilidades tan especializadas los resultados suelen ser mejores que con los programas de edición gráfica.

Ejemplos de optimizadores gráficos:

- [WebGraphics Optimizer](#)
- [ProJPG, GIF Imantion](#)

Y con versiones Online:

- [JPG - GIF Crunchers](#)
- [GIF Wizard](#)



Photoshop es una herramienta excelente para optimizar ficheros. Viendo varias copias podemos elegir la más adecuada.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Icono en favoritos

En este artículo vamos a aprender a colocar un icono en el menú de favoritos, de modo que cuando alguien guarde la página en sus favoritos se vea la página con el icono que nosotros deseemos, en lugar del icono de Internet Explorer. El icono de nuestra página también se podrá ver en la barra de direcciones del explorador, siempre a partir del momento en que un usuario guarde la página en favoritos.

Es un efecto muy sencillo y vistoso, pero tiene un serio inconveniente: solo funciona en Internet Explorer versión 5. No funciona en Netscape ni en versiones distintas de Explorer.

Cómo conseguir un icono

Un icono es un archivo especial, que tiene extensión .ico. Los editores de gráficos habituales no trabajan con este formato, por lo que se requerirá del uso de una herramienta especial para la creación y tratamiento de la imagen que vamos a utilizar como icono.

Una popular herramienta útil para trabajar con iconos es [Microangelo](#). Permite la creación de gráficos que luego se pueden guardar como .ico. También permite convertir a iconos gráficos en formato GIF.

Donde colocar el icono

Para conseguir nuestro objetivo tenemos dos posibilidades. La primera se trata de colocar un icono a todo un sitio web entero. Con la segunda opción podremos asignar un icono en una página de manera independiente.

Colocar un icono en un sitio, de manera global

Para ello debemos conseguir un archivo ico y darle el nombre "favicon.ico". Este archivo lo colocaremos en el directorio raíz del dominio. Cuando se introduzca en favoritos cualquier página del sitio se verá el icono seleccionado.

No siempre tenemos acceso al directorio raíz del dominio. Imaginemos que publicamos una web en Geocities, en este caso nunca tendremos acceso al directorio raíz, [www.geocities.com](#), sino que tendremos acceso al directorio que nos hayan asignado, [www.geocities.com/mipagina](#). Para este caso, tenemos una segunda manera de insertar el icono.

Colocar un icono específico en una página

En este caso tenemos que colocar una etiqueta en la cabecera del documento HTML (entre y). Esta etiqueta la tenemos que colocar en cada página que queramos asociar con un icono.

```
<link rel="SHORTCUT ICON" href="myicono.ico">
```

Por supuesto, en el atributo href de la etiqueta anterior, se ha de colocar el nombre del icono que queremos utilizar y la ruta para acceder a él. En el caso del ejemplo anterior se supone que un archivo llamado myicono.ico estará en el mismo

directorio que la página web.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

GIFs animados

No sabes hacer un círculo con el PaintBrush? no te preocupes, igualmente puedes hacer magníficas animaciones. Eso si, necesitarás la ayuda de alguna aplicación que haga algo de trabajo por ti. En la sección de [recursos](#) puedes encontrar lo que necesitas. Para la realización de los ejemplos hemos utilizado el Animator Shop 2 que acompaña al Paint Shop 6.0.

Lo mas fácil

El primer paso es tener una idea, cosa nada fácil. Se pueden hacer infinidad de maravillas, lo mejor es buscar buenos resultados con poco trabajo. Hay que recordar siempre las limitaciones de tamaño en internet, con lo que se debe trabajar siempre con poca profundidad (cantidad de colores) y tamaños reducidos. También es mejor usar un buen programa que optimize el tamaño de la animación.

Para el que realmente tenga problemas para hacer un círculo puede crear una animación con cualquier imagen GIF, aplicandole alguno de los efectos que ofrecen las aplicaciones. Un ejemplo de la aplicación de un simple efecto a una simple imagen, que se puede hacer con cuatro clicks:



Tus propias animaciones

Mucho más interesante es el realizar uno mismo su propia animación para eso solo hay que seguir unos sencillos pasos. Primero debes determinar el tamaño de la imagen animada, cada uno de los fotogramas debe respetar ese tamaño. También debes encontrar el número mínimo de fotogramas que necesitas para crear el realismo deseado, este número no suele ser muy alto. Debes seguidamente realizar cada uno de los fotogramas, la mejor manera es hacer pequeños cambios en cada uno de ellos. Después los reunimos en un programa de animación y los montamos. Una muestra de lo que se puede realizar con unas decenas de clicks:



Realizada simplemente girando cada una de las letras de forma diferente. Sobre deciros que para completar el ciclo correctamente es necesario que el último

fotograma sea similar al primero. Gracias a la baja profundidad (16 colores) y a que utilizamos solo cuatro fotogramas la animación solo ocupa 988 bytes!

Ultimas sugerencias

Esta animación ha necesitado unos centenares de clicks. Ha estado realizada en dos partes: la animación de las bandas laterales y la de la imagen interior. Aún con la mínima profundidad (2 colores) el archivo ocupa 11k ya que esta compuesto de 27 fotogramas.



Hay otros importantes factores sobre los que podemos jugar, al menos si nuestro software lo permite. Uno es la duración de los fotogramas, podemos ralentar o acelerar las animaciones. Pero cuidado, porque fotogramas lentos disminuyen la fluidez de la animación, recordamos que lo importante es que el ojo no llegue a "ver" cada uno de los fotogramas por separado.

Otra posibilidad es la de no repetir indefinidamente la animación, sino pasarla una sola vez o un numero determinado de veces. Muy útil como inicio de vuestras páginas. Por último decir que las animaciones también se pueden utilizar como fondos, estas animaciones deben ser muy sutiles para no cargar demasiado la página.

Programas para crear GIFs animados

Hay multitud de programas para la creación de gifs animados. Podemos hacernos una idea del número de aplicaciones existentes visitando el [directorio de animación de imágenes del sitio de descargas Tucows](#), donde hay muchas utilidades para animación en general y específicas para gifs animados.

De todos modos, aquí tenemos una serie de recomendaciones de programas.

- Photoshop, a partir de su versión 5.5, incorpora una utilidad complementaria llamada Image Ready para hacer, entre otras cosas, gifs animados.
<http://www.adobe.es/products/photoshop/>
- el Xara3d, es bueno para hacer titulos sencillos, en 3d y animados.
<http://www.xara.com/products/xara3d/>
- Tambien tenes el "unlead gif animator", para algo mas elaborado.
<http://www.ulead.com/ga/>
- Con el "firewoks", podes hacer gif animados de muy buena resolucion y poco peso. <http://www.macromedia.com/es/software/fireworks/>
- Si necesitas movimientos mas complejos podes hacerlos en Flash y lo exportas como .gif animado.
<http://www.desarrolloweb.com/articulos/338.php>

Nota: Esta referencia a los programas para hacer gifs animados se la debemos a Miguel Angel Alvarez y Quo Vadis Pepus. Sin ella, este artículo no quedaría completo.

Informe de **Luis Vilaroig**
Mail: lvilaroig@desarrolloweb.com

Los colores y HTML

En la composición de webs juegan un papel muy importante los colores. Se indican en valores RGB, es decir, que para conseguir un color cualquiera mezclaremos cantidades de Rojo, Verde y Azul.

Los valores RGB se indican en numeración hexadecimal, en base 16. (Los dígitos pueden crecer hasta 16. Como no hay tantos dígitos numéricos se utilizan las letras de la A a la F.



Tabla de color

0=0	4=4	8=8	C=12
1=1	5=5	9=9	D=13
2=2	6=6	A=10	E=14
3=3	7=7	B=11	F=15

Para conseguir un color, mezclaremos valores de esta manera:

RRGGBB

Donde cada valor puede crecer desde 00 hasta FF.

Ejemplo: Cómo se cambiaría la fuente para escribir en rojo:

```
<font color="#FF0000">Rojo</font>
```

Al atributo **color** le damos un valor RGB en formato hexadecimal. El carácter # se coloca al principio de la cadena.

Otros colores:

Naranja #FF8000
Verde turquesa #339966
Azul oscuro #000080

Colores compatibles en todos los sistemas


Como las páginas web las tienen que ver todos los usuarios, y los sistemas que utilizan para entrar son distintos, hay que utilizar colores compatibles con la paleta de todos ellos.

La forma de conseguir esto es limitando nuestros colores a los que se pueden conseguir utilizando la siguiente norma:

Utilizaremos siempre estos valores:
00
33
66
99
CC
FF

Ejemplos: #3366FF #FF9900 #666666

Se consiguen los colores siguientes:

 #000000	#000033	#000066	#000099	#0000CC	#0000FF
#003300	#003333	#003366	#003399	#0033CC	#0033FF
#006600	#006633	#006666	#006699	#0066CC	#0066FF
#009900	#009933	#009966	#009999	#0099CC	#0099FF
#00CC00	#00CC33	#00CC66	#00CC99	#00CCCC	#00CCFF
#00FF00	#00FF33	#00FF66	#00FF99	#00FFCC	#00FFFF
#330000	#330033	#330066	#330099	#3300CC	#3300FF
#333300	#333333	#333366	#333399	#3333CC	#3333FF
#336600	#336633	#336666	#336699	#3366CC	#3366FF
#339900	#339933	#339966	#339999	#3399CC	#3399FF
#33CC00	#33CC33	#33CC66	#33CC99	#33CCCC	#33CCFF
#33FF00	#33FF33	#33FF66	#33FF99	#33FFCC	#33FFFF
#660000	#660033	#660066	#660099	#6600CC	#6600FF
#663300	#663333	#663366	#663399	#6633CC	#6633FF
#666600	#666633	#666666	#666699	#6666CC	#6666FF
#669900	#669933	#669966	#669999	#6699CC	#6699FF
#66CC00	#66CC33	#66CC66	#66CC99	#66CCCC	#66CCFF
#66FF00	#66FF33	#66FF66	#66FF99	#66FFCC	#66FFFF
#990000	#990033	#990066	#990099	#9900CC	#9900FF
#993300	#993333	#993366	#993399	#9933CC	#9933FF
#996600	#996633	#996666	#996699	#9966CC	#9966FF
#999900	#999933	#999966	#999999	#9999CC	#9999FF
#99CC00	#99CC33	#99CC66	#99CC99	#99CCCC	#99CCFF
#99FF00	#99FF33	#99FF66	#99FF99	#99FFCC	#99FFFF
#CC0000	#CC0033	#CC0066	#CC0099	#CC00CC	#CC00FF
#CC3300	#CC3333	#CC3366	#CC3399	#CC33CC	#CC33FF
#CC6600	#CC6633	#CC6666	#CC6699	#CC66CC	#CC66FF
#CC9900	#CC9933	#CC9966	#CC9999	#CC99CC	#CC99FF
#CCCC00	#CCCC33	#CCCC66	#CCCC99	#CCCCCC	#CCCCFF
#CCFF00	#CCFF33	#CCFF66	#CCFF99	#CCFFCC	#CCFFFF
#FF0000	#FF0033	#FF0066	#FF0099	#FF00CC	#FF00FF
#FF3300	#FF3333	#FF3366	#FF3399	#FF33CC	#FF33FF
#FF6600	#FF6633	#FF6666	#FF6699	#FF66CC	#FF66FF
#FF9900	#FF9933	#FF9966	#FF9999	#FF99CC	#FF99FF
#FFCC00	#FFCC33	#FFCC66	#FFCC99	#FFCCCC	#FFCCFF
#FFFF00	#FFFF33	#FFFF66	#FFFF99	#FFFFCC	#FFFFFF

Por último, te ofrecemos unos archivos con programillas en JavaScript que te pueden ayudar a encontrar los colores adecuados. Entre otros, podrás encontrar el script que genera la tabla de arriba.

 [Colores.zip](#) 16Kb

Informe de **Miguel Angel Alvarez**
 Director desarrolloweb.com
 Mail: eugim@desarrolloweb.com

Definiciones de pantalla

Cuando trabajas a **distintas definiciones**, como en la web, debes elegir el

público objetivo de la página y construirla para que ese público la vea bien, pero no te debes olvidar de los demás, de modo que estos también la visualicen, a ser posible, sin ningún problema.

El público objetivo actual de cualquier página debería de ser el que tiene la definición de **800x600**, ya que la mayoría de usuarios utilizan esta resolución de pantalla. También navega por Internet mucha gente a **640x480**, pero cada vez son menos el número de personas que utilizan esta definición. Para el caso de **1024x780**, esta definición es demasiado grande y exclusiva de usuarios con buenas máquinas, configuradas de manera avanzada, como deja fuera muchas personas no debemos utilizarla todavía, además, estos usuarios podrán ver también la página a definición menor sin demasiado perjuicio. Una última consideración sería que si la página a desarrollar es muy corporativa, de modo que pertenezca a una empresa importante y se tenga que ver bien en todas las máquinas posibles, sería razonable utilizar la definición de 640x480, pues esta nos asegura que todo el mundo podrá ver bien el web.

Una vez escogido el público objetivo, se han de componer las páginas y las imágenes para se vean bien en la definición de pantalla que utilicen estos, de modo que nunca salgan las feas barras de desplazamiento horizontales. Para ello calcularemos el tamaño de los elementos de la página concienzudamente.

Otra cosa que se puede hacer es aplicar a los elementos de la página los tamaños (atributo width) utilizando porcentaje, de este modo, se ajustarán al tamaño de la definición del usuario automáticamente. Sin embargo, las imágenes no soportan el tamaño en porcentajes, es decir, no podemos ajustarlas así automáticamente, y en cualquier caso, no desearemos que la imagen se deforme al alterar artificialmente sus tamaños a través de estos atributos. Así pues, para el caso de las imágenes seguiremos en la necesidad de crearlas sin que exceda su tamaño horizontal la definición horizontal de público objetivo.

El que firma este artículo siempre utiliza tablas para maquetar las páginas que realiza. El método utilizado es el siguiente:

1. Hablamos de un diseño con tablas porque casi siempre es más adecuado que un diseño con frames, pero esto es otra discusión.
2. **Crear una tabla al principio de la página, que incluya toda la página**, y le asignamos el tamaño en pixels. Dependiendo de la resolución del tarjet de audiencia, el tamaño de la tabla variará, pero siempre serán 20 pixels menos que la definición objetivo. Estos pixels sobrantes se utilizan para las barras de desplazamiento verticales. Por ejemplo, si diseñamos para una resolución de pantalla de 800x600, el tamaño de la tabla será de 780 pixels.

También se pueden crear las tablas en porcentajes, pero esto hará que la página se estire y se encoja dependiendo de la definición. No es un efecto muy deseable para los diseñadores, porque habrá que conseguir que la página se vea bien en todas las definiciones, y seguro que no se ve tan bien como lo que nosotros teníamos con la definición en la que estábamos diseñando, además de significar un esfuerzo extraordinario.

3. **Incluimos en la etiqueta <BODY> de la página los atributos: `topmargin=0 leftmargin=0 marginheight=0 marginwidth=0`**. Estos atributos sirven para eliminar los márgenes de la página y que las tablas se sitúen ocupando todo el espacio de la página. Los dos primeros atributos son para Internet Explorer, los dos siguientes son para Netscape Navigator, tenemos que poner los cuatro para asegurarnos que se vea bien en todos los navegadores más importantes.

4. Si deseamos incluir **márgenes en las páginas**, podemos jugar con los atributos **cellspacing y cellpadding**, en la tabla principal, para conseguirlos. En caso de que no nos guste este método podemos incluir **celdas adicionales en la tabla principal, que sean transparentes** (no meterles contenido), a las que les asignamos los tamaños deseados para los márgenes. El método que utilizo habitualmente es este último y casi siempre pongo los atributos cellspacing y cellpadding a cero.

Eso es todo, recuerda que siempre puedes ver el código fuente de las páginas web que te gustan, para ver cómo lo han hecho otros diseñadores.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Elegir alojamiento gratuito

Publicar una página web es el objetivo de la mayoría de las personas que visitan desarrolloweb.com. Para conseguirlo lo más imprescindible es **tener un servidor donde colocarlas**, accesible desde Internet, aparte de construir las propias páginas.

Dependiendo de cuál sean nuestras intenciones y deseos para el futuro de nuestra página deberemos elegir **un proveedor gratuito o uno de pago**, Será recomendable optar por uno de pago, en un proveedor seleccionado, si tratamos de hacer una web de una empresa o una que pretenda ser muy visitada, una web que ofrezca unos servicios y prestaciones más elevados tanto al administrador como a los visitantes. En este caso, conviene leer mejor el [reportaje Cómo elegir un proveedor de pago](#),

Si por el contrario, deseamos publicar un trabajo con menores prestaciones, o simplemente no queremos gastar dinero en el hosting, será indicado optar por un **servicio de alojamiento gratuito**. Por suerte, existe una gran variedad de sitios que ofrecen a sus visitantes la oportunidad de publicar sus paginas personales o de empresa de manera gratuita y muchas de estas opciones tienen muy buena calidad.

Donde elegir un espacio

En un principio, el mejor espacio para publicar una web que podemos elegir puede ser el proporcionado por **nuestro proveedor de acceso a Internet**. Muchas veces nuestro proveedor, a la vez que ofrece la conexión a Internet, también ofrece un espacio para publicar, por el mismo precio que el acceso. Esta opción tiene sus ventajas:

- Suelen estar libres de publicidad
- En muchos casos, existen menos limitaciones por el tipo de contenidos
- Casi siempre tienen buena velocidad.

El único inconveniente a destacar -aparte de los problemas puntuales que tenga cada proveedor de acceso- es que **abandonemos algún día el proveedor y nos retiren las páginas publicadas**, pero esto se puede evitar teniendo un servicio de redirección que nos de una URL (dirección de web) para toda la vida, que

redireccione a la página que esté activa en ese momento. Para ver servicios de redirección podemos [consultar nuestro buscador](#).

En el caso de que nuestro proveedor no ofrezca servicio de publicación de webs o no nos guste el servicio que ofrece, tenemos la opción de publicar las páginas en alguna **web que ofrezca espacio gratuito**. Este tipo de servicio es muy popular, existen muchas ofertas en Internet, algunas de ellas mejores de las proporcionadas por los proveedores de acceso, incluso por los proveedores de alojamiento.

En qué debemos fijarnos

Vamos continuar explicando algunos puntos en los que fijarnos para poder elegir un proveedor gratuito. Dando algunos ejemplos de sitios que ofrecen espacio con buenas prestaciones.

Recursos permitidos

Existen muchos sitios donde publicar una página, se diferencian fácilmente por los recursos que ponen a nuestra disposición para construirla. Cuando hablamos de recursos nos referimos a la **cantidad de herramientas y programas que nos dejan utilizar en nuestra web**. Ejemplos de recursos son: estadísticas, contadores, direcciones web cortas, mail gratuito, CGI, bases de datos... Los recursos se hacen necesarios cuando construimos una web avanzada, con secciones dinámicas como un buscador, un portal, o en general, una página que se alimenta de cualquier base de datos o programa para mostrar su contenido.

En la gama de recursos que ofrecen estos sitios gratuitos es muy variada. Algunos sitios casi no ofrecen ningún recurso, simplemente podemos publicar la página, y en otros sitios ofrecen más servicios que en algunos proveedores de alojamiento de pago.

Al final del reportaje podremos ver una comparativa sobre la cantidad de recursos ofrecida por algunos sitios proveedores de espacio gratuito para publicar webs.

Velocidad del servidor

Para elegir un servicio suficientemente veloz tenemos la oportunidad de **probar la rapidez con que bajan las páginas de alguna web que esté alojada en ese servidor**. Es decir, visitamos una web de Geocities.com y vemos lo rápido que llega, una web de Galeon.com y vemos la rapidez, comparando nosotros mismos. En un último caso podemos registrarnos nosotros mismos y probar la velocidad del espacio con nuestros propios contenidos. No pasa nada por registrarse en un sitio, probar cómo funciona y si no nos gusta simplemente nos vamos a otro, es gratis.

Para poder comprobar la velocidad de varios sitios de alojamiento gratuito hemos creado un web y lo hemos subido a servidores distintos. Con los siguientes enlaces podemos comprobar en qué servidor es más rápido para nosotros:

Galeón: <http://www.galeon.com/ipaginate>

Iespana: <http://ipaginate.iespana.es>

Geocities: <http://www.geocities.com/michel98.geo/testvelocidad/index.html>

Freeservers: <http://quintointento.freeservers.com>

Metropoli2000: <http://desarrolloweb.metropoli2000.net>

Un dato a considerar es que, dependiendo de dónde está situado físicamente el servidor que alojará nuestras páginas, la velocidad será distinta para nosotros. Es decir, un español ve más rápido un servidor español, pero un americano navegará con más velocidad si el servidor está en América. Esto, se lleva al límite cuando es nuestro proveedor de acceso el que nos ofrece el espacio, en este caso, seremos los que navegen más rápido por esa web, por estar tan próximos los dos sitios, tu acceso a Internet y el servidor de tu página.

Publicidad insertada en las páginas

También se pueden **clasificar los sitios para publicar en Internet por la publicidad que insertan**. Los sitios de espacio gratuito suelen subvencionarse mediante la inserción de anuncios en las páginas de los usuarios del servicio. Los tipos de publicidad que podemos encontrar son los siguientes:

- **Publicidad en un frame inferior.** Es tal vez la más agresiva de las publicidades que pueden hacer, pues el banner permanece siempre visible en la página y es un poco feo para las páginas.
- **Publicidad dentro de la página, utilizando el banner superior.** Es el método de inserción de banners más habitual. Como desventaja tiene que el banner se exhibe en la propia página, modificando nuestra web.
- **Publicidad en una ventana aparte.** Esta publicidad aparece en una ventanita aparte del explorador. Como ventaja está que no utiliza el espacio de nuestra página. Como desventaja contamos que estas ventanas son muy molestas para muchos usuarios.
- **Publicidad en la propia página, en un grafiquito flotando en la propia página.** Es muy interesante, pues ocupa muy poco espacio comparado con otros mecanismos y no molesta en exceso.
- **Sin publicidad,** sin duda el mejor de los casos. Lo que no nos asegura que el sitio no ponga publicidad algún día y deje de ser tan interesante. Actualmente, muy pocos sitios ofrecen espacio gratuito sin colocar algún tipo de publicidad.

Hay muchos tipos de publicidad insertada habitualmente, como se puede ver. Eso no quita que algunos sitios empiecen a utilizar varios sistemas a la vez.

Los ejemplos de sitios que estamos evaluando tienen la siguiente forma de insertar la publicidad en las páginas de los usuarios:

Galeon: En el momento de escribir este reportaje no colocaban banners en las páginas alojadas en su servidor. Sin embargo, actualmente si colocan publicidad en una capa que acompaña tu navegación y permanece siempre visible.

Metropoli 2000: Colocan un banner en la parte de arriba de la página. Es una opción interesante, porque reparten los ingresos por publicidad con el dueño de la página.

Iespana: Los banners de este sitio se colocan en un frame en la parte inferior de la página. Pero con un simple aviso podemos pasar cambiar esto para que la publicidad salga en una ventana aparte.

Geocities: Colocan un banner cuadrado que se puede incluso borrar de la web a base de clicks. El problema es que este banner se coloca con Javascript y en determinados navegadores da errores, lo que es peor incluso que un banner grande. En mi opinión un fallo de Javascript es algo bastante molesto y en mi Internet Explorer 4 fallan casi todas las páginas de Geocities.

Freeservers: Colocan un banner en la parte superior de la página, con enlaces a otras partes de Freeservers. Además, también colocan publicidad en ventanas a parte. Todos estos males se pueden eliminar si el usuario paga una cuota mensual.

Por manera de colgar los archivos

Existen diferentes maneras de subir los archivos. Por ejemplo, **los podemos subir con FTP o por medio de una página donde existe un formulario y donde podemos seleccionar los archivos que deseamos subir**. En la descripción de características del alojamiento suelen indicar acceso por FTP cuando esta permitido realizar este tipo de actualizaciones.

Para los **principiantes** será más cómodo subir las páginas por medio de **un formulario de una web**, pero para las personas **experimentadas**, a la larga, les será más sencillo y tendrá menos trabajo el realizar las actualizaciones del sitio **via FTP**.

Como ventaja de subir archivos con un formulario tenemos que no será necesario disponer de un programa de FTP ni tampoco saber utilizarlo. Como ventaja del FTP destacamos la rapidez de subida de archivos, ya que podemos seleccionar varios archivos de una sola vez y no tenemos que esperar a que se cargue la página web para que tengamos acceso al formulario, subir un directorio en una sola acción y otra mejora importante, que consiste en que podremos utilizar el programa con el que diseñamos las páginas web (tipo Dreamweaver o Hometown) para acceder al servidor y actualizar los contenidos automáticamente.

Nuestro consejo es aprender a utilizar FTP, si es que deseamos trabajar de manera más profesional y ahorrar tiempo a la larga. Ver el [siguiente enlace para conocer un programa de FTP y tener acceso a un tutorial para aprender su uso](#).

Si queremos aprender a subir archivos, tenemos también un [artículo muy interesante](#).

Comparativa y conclusión

Aquí podemos ver una **tabla con las principales características** de los sitios que ofrecen alojamiento gratuito destacados en este reportaje.

	Geocities	Metropoli 2000	Iespana	Galeon	Freesevers
Espacio	15M	Ilimitado	Ilimitado	Ilimitado	12M
Correo	SI	Ilimitado	SI	SI	Sólo redirección
Estadísticas	SI	SI	SI	NO	SI
Dominio	NO	Pagando	Pagando	NO	Pagando
Subdominio	NO	SI	SI	NO	Varios a elegir
Páginas dinámicas	NO	PHP	PHP	NO	NO
Bases de datos	NO	MySQL	MySQL	NO	NO
Multimedia	Típica	Típica	Real Networks	Típica	Típica
CGI instalados	Alguno	Numerosos	Numerosos	NO	Algunos
CGI propios	NO	SI	NO	NO	NO
Herramientas componer y publicar	SI	?	SI	SI	SI
Acceso por FTP	SI	SI	SI	SI	NO
Publicidad	Cuadrado flotante	Banner superior	Frame inferior o	Cuadro flotante y	Banner superior

Transferencia*	Sin dato. limitada?	ingresos a compartir Ilimitada	ventana aparte Ilimitada	popus abusivos Sin dato. ilimitada?	512 MB/mes
Observaciones	Tiene errores javascript	Solo para contenidos de calidad	Para todos los públicos, buenas herramientas	La única sin publicidad	Nos costó mucho registrarnos

***Transferencia:** Es la cantidad de información que recibes y mandas cada mes el servidor cuando le piden tu página. Empiezan a haber servicios gratuitos que limitan la transferencia porque al fin y al cabo es lo que cuesta dinero del hecho de ofrecer espacio gratuito y poco a poco se cierra el grifo de la Internet gratuita. Tener cuidado con este detalle, porque puede que el alojador gratuito elegido pueda ser inútil debido a que tenga mucho éxito tu web y tenga limitado el tráfico.

Los Sitios que ofrecen **mayores recursos para hacer una página web son Iespana.es y Metropoli2000.net**, con la principal diferencia de que

En **Metropoli2000 aceptan webs con contenidos de calidad**, sin despreciar al diseño. Es decir, necesitas tener una web en Internet previamente publicada y estos te dicen si es de suficiente calidad para poder publicarla en su servidor. Sin embargo, ellos te remuneran por los ingresos de publicidad, aunque actualmente parece ser que el mercado les ha hecho desistir de esta posibilidad.

Por lo tanto **Iespana es la más accesible**, por permitir todo tipo de contenidos, incluso comerciales o de adultos. Además nos permiten programar en PHP 4 con base de datos MySQL, lo que hace muy interesante para aquellos desarrolladores que deseen programar aplicaciones de servidor.

Galeón era un lugar interesante, gracias a su ausencia de publicidad, aunque ahora si que ponen publicidad y cada vez más abusiva. Actualmente saltan ventanas secundarias por todas partes y encima los cuadros flotantes son cada vez más grandes.

Freeservers no nos funcionó del todo bien, sobretodo en el registro, que tuvimos que intentarlo durante un par de días en un total de 5 ocasiones antes de conseguir ver la hoja final de registro.

Para mi, **Geocities es la peor opción**. Es poco claro, no ofrecen muchos recursos y la publiciada es en ocasiones más molesta que en otros sitios, llegando incluso a provocar errores javascript en el momento de la redacción de este artículo.

Direcciones de espacio gratuito

Iespana: <http://www.iespana.es>
 Metropoli 2000: <http://www.metropoli2000.net>
 Freeservers: <http://www.freeservers.com>
 Geocities: <http://www.geocities.com>
 Galeón: <http://www.galeon.com>

Mirar también:
[Otras opciones de alojamiento gratuito en nuestro buscador](#)

Informe de **Manuel Alamar**
 Mail: manuelalamar@guiarte.com

Cómo elegir un alojamiento

Para publicar una página en Internet debemos colocarla en un servidor. Los servidores son ordenadores conectados permanentemente a la Red que envían las páginas cuando los exploradores la piden.

Antes de nada vamos a diferenciar entre dos formas de alojar una web, en un servidor gratuito o en un servidor de pago. Los servidores de pago suelen tener mayores prestaciones y recursos disponibles que los gratuitos. En este reportaje vamos a tratar de orientar a las personas que tienen que contratar un servicio de pago, más indicado para negocios y páginas que pretendamos que sean populares.

Si no deseamos pagar por nuestro alojamiento podemos consultar el reportaje [Cómo elegir un alojamiento gratuito](#).

Definir los recursos

Para encontrar un alojamiento para nuestra página primero nos tenemos que plantear los objetivos que queremos cumplir y los recursos que necesitaremos. Según sea el proyecto que vamos a publicar necesitaremos más o menos personalidad propia o determinados recursos como acceso a bases de datos, estadísticas, etc.

El primer paso podría ser decidir si vamos a necesitar un dominio propio para nuestro proyecto. Un dominio propio le dará más personalidad al sitio, se asociará mejor con el nombre de la empresa, a la vez que se hace más accesible la página. Con el reducido precio que tienen actualmente los dominios sería muy indicado adquirir uno para nuestra página. En muchos casos al contratar un espacio para alojar la web nos van a ofrecer directamente nuestro propio dominio, si no es así, lo podemos solicitar expresamente.

Lo más habitual es que en el proveedor donde alojemos la página se encargue también del registro del dominio, sin prácticamente otro coste adicional que las tasas del NIC, 30\$ aproximadamente. Para conocer más sobre los dominios y su registro se puede [consultar un reportaje](#) también publicado en desarrolloweb.

Aparte del dominio existen otros recursos podemos necesitar, como:

- Programación en lenguajes específicos
- Acceso a bases de datos
- Administración de las DNS...

Deberemos definir los recursos a utilizar y buscar un proveedor que los acepte todos. Si estás proyectando una página sencilla, probablemente no te preocupará el número de recursos avanzados que posea tu servidor, pero siempre será interesante evaluar otros recursos más básicos como:

- El número de direcciones de correo que te van a proporcionar
- Los CGI's que tenga instalados, como contadores, envío de formularios por correo electrónico, etc.
- Estadísticas...

Velocidad

Es una pena que no exista un método realmente fiable para controlar uno de los aspectos que más preocupa a todos los que buscamos un alojamiento. Para conocer la calidad el servicio que nos va a ofrecer el proveedor deberíamos preguntarles a ellos sobre sus líneas y la capacidad que tienen de transferencia con Internet. Pero no solo eso sino también preguntarles por la saturación que tienen sus líneas. Pueden ser muy potentes pero estar super-utilizadas. Como los proveedores no van a revelar estos detalles tal como a nosotros nos gustaría, lo más probable es que solo nos quede la posibilidad de probar la velocidad por la práctica. Si navegamos por el sitio de la empresa que estamos evaluando y comprobamos la cantidad de información (Kb) que recibimos por segundo podremos saber cómo va de rápido la página.

Otro factor que también condiciona la velocidad es la cercanía del servidor. Ten en cuenta que si estás en España, un servidor situado en tu país será más rápido para ti, en la misma condición de líneas, que uno que esté en Estados Unidos. Cuanto más lejos tenga que viajar la información, tarda más. Pero un detalle, si un usuario de Colombia visita una página que está en un servidor español, para él tu servidor será más lento que uno de su país.

Transferencia

Otro aspecto que tenemos que evaluar a la hora de contratar un espacio para nuestra web es la cantidad de megas de transferencia mensual que el proveedor permite realizar desde nuestro dominio hacia fuera. Es decir, las páginas web que manda el servidor tienen un peso en Kbytes y el proveedor los va contando, cuando pasamos el límite que le han asignando al dominio nos cobran el exceso según un precio.

Así pues, a la hora de contratar un dominio merece la pena enterarse sobre este aspecto y evaluar también las condiciones que nos ofrecen. Si encontramos un servidor que ofrece transferencia ilimitada no nos tiene que decir tampoco mucho de él, el control de la transferencia les permite ajustar su servicio a unos niveles de calidad superiores. Además, en la mayoría de los casos no nos tendrá que preocupar el superar las tasas de transferencia pues las megas que transfiere una web normal nunca superará los niveles propuestos, a no ser que se trate de un portal o una página muy visitada.

Precio

El último factor a considerar, y el más fácil de comparar, es el precio. Lo importante es comparar la calidad del servicio y el precio del mismo. Seguro que podrás realizar esta comparación sabiendo los datos que has de evaluar.

Buscador de Alojamiento

En DesarrolloWeb podemos ofreceros una gama de [alojamientos de las más altas prestaciones](#). Desde el registro de dominios o alojamientos en Unix y NT con los recursos más avanzados, hasta servidores privados para administrar tu página y la de todos tus clientes.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Dominios y cómo registrarlos

Un dominio es una forma sencilla de identificar un ordenador en Internet de manera única, a partir del cual se encontrarán las páginas pertenecientes a la institución que lo posee.

Los dominios tienen un nombre y una terminación que indica su actividad o procedencia territorial. Por ejemplo, **yahoo.com**: **yahoo** es el nombre y **.com** expresa el ámbito de esa página, comercial en este caso.

Escoger bien el nombre del dominio es fundamental, si este tiene gancho podremos ser fácilmente recordados e identificados en Internet. A veces no es tan sencillo realizar una buena elección del nombre del dominio pues puede que los que nos gusten ya hayan sido registrados por otras personas.

En muchos casos, el registro de los dominios ha sido realizado como un auténtico pillaje, solicitando los usuarios de Internet miles de nombres con el **único objetivo de poseer ese dominio con gancho**, pensando en futuras compensaciones, y sin presentar ninguna información en sus páginas web. Los nombres de las empresas importantes son los que más han sufrido este pillaje, que ha obligado a las empresas a **comprar los dominios "raptados" por un precio astronómico** a sus dueños.

Tipos de dominios

En Internet existen varios tipos de terminaciones de dominios o, mejor dicho, **dominios de primer nivel**. Estos son los.com, .org, .es, etc.

Como decíamos, los dominios de primer nivel **indican el ámbito al que pertenecen**, hay principalmente dos grupos, **genéricos y territoriales**.

Dominios genéricos

Son dominios que se otorgan a nivel internacional, para empresas y personas de todo el mundo. Los vamos a enumerar aquí, indicando el tipo de institución al que van dirigidos.

.com Para empresas o en general para cualquier web que tenga carácter comercial. En un principio, quería decir que ese dominio que se trataba de una **compañía** estadounidense, pero en la práctica cualquiera ha tenido acceso a estos dominios que se han hecho muy populares y los preferidos para cualquier tipo de fin.

.net Indica una **red** en Internet, la de un proveedor de servicios por ejemplo. Una opción que a la larga también se ha convertido en válida para cualquier tipo de propósito.

.org Destinado para **organizaciones**, asociaciones, fundaciones y demás entidades muchas veces con fines benéficos o si ánimo de lucro.

.gov Es para las páginas del **gobierno** de los Estados Unidos.

.edu Reservado para las instituciones relativas a la **educación**, pero solo las de los Estados Unidos.

.mil Se utiliza para instituciones **militares** de los estados unidos.

.int Que pertenece a la **Unión Internacional de Telecomunicaciones**, y en el que se pueden encontrar organismos que se hayan creado con acuerdos internacionales, como las Naciones Unidas.

Nota: Actualmente, se han aprobado 7 dominios nuevos. Se pueden ver unas [descripciones de estos en este artículo](#).

Hay un organismo llamado **NIC** que es el que se encarga de **regular el registro de los dominios** a nivel mundial. Se encarga de indicar para que se utiliza cada dominio, quien está autorizado a registrarlo, y quien puede ser el registrador. Esta entidad delega en otras para desempeñar todo el trabajo de organización que conlleva la administración de los dominios, sobretodo en los distintos países, como más adelante se verá.

Para registrar un dominio

Anteriormente, el registro de dominios sólo lo podía realizar una empresa llamada Network Solutions, que mantuvo el monopolio hasta el verano de 1999. Actualmente se encuentra liberalizado este mercado y **existen muchas más empresas registradoras de dominios**. ([Ver el listado](#)) De todos modos, a través de estas empresas operan muchos más intermediarios y encontrar un registrador cercano a nosotros puede ser muy sencillo.

Sólo los dominios del tipo **.com .net y .org** (y ahora también los .info .biz y .edu) se encuentran al alcance de cualquier persona. Para registrarlos podemos acceder a las páginas de las empresas que están capacitadas para ello. En DesarrolloWeb.com podemos ayudaros también en esta tarea.

En nuestro [servicio de alojamiento](#) puedes [solicitar un dominio y registrarlo](#) para cuanto tiempo desees. Además, en la parte de la derecha puedes ver un formulario donde escribir y buscar el dominio que desees. Posteriormente, si es que tu dominio está libre, podrás registrarlo a través de nuestras páginas. En estas páginas debemos realizar una búsqueda para saber si se encuentra disponible el dominio que deseamos. Una vez hemos comprobado que no pertenece ya a ninguna persona podemos pasar a su registro, que se realiza a través de unos formularios online en esas mismas páginas.

Si vamos a colocar unas páginas web en el dominio y vamos a contratar los servicios de hosting en algún **proveedor**, puede ser aconsejable que este **sea el encargado de la labor de registro**. Lo hará a través de una de esas empresas capacitadas para ello (señaladas arriba) y, generalmente, sin un coste adicional, o en el caso de haberlo, no será muy elevado. (probablemente el mismo precio que tenga trasladar ese dominio a sus servidores)

Hay unos costes relacionados al registro de los dominios, costes que no se pueden evitar y que no se los queda el proveedor, sino que son destinados al NIC. Para los dominios **.com, .net y .org**, el coste del registro es de **35 dólares**. Esta tasa permite mantener el dominio durante 1 año. Posteriormente hay que pagar 35 dólares por año para seguir manteniendo el nombre.

Dominios territoriales

También existen **dominios de primer nivel que indican el territorio** de origen de la página. Estos dominios solo se le otorgan a **empresas o personas de los países relacionados** con el dominio.

Como ejemplos de dominios territoriales podemos señalar **.es** para España, **.fr** para Francia, **.mx** para México...

El registro de los dominios territoriales es regulado en base a unas normas específicas para cada país. Los encargados de crear estas normas para el registro son los distintos **delegados del NIC de cada país**. De este modo, el **ES-NIC** (www.nic.es) es el encargado en España, mientras que **MX-NIC** (www.nic.mx) es el de México, o **AR-NIC** (www.nic.ar) el de Argentina.

España, por ejemplo, las normas para el registro de un dominio .es son bastante restrictivas, y no todo el mundo puede registrarlos. En concreto para acceder a estos debemos **ser una empresa y nuestro nombre debe de ser igual al del dominio que queremos registrar**, o muy parecido. También es posible el registro si se posee una marca registrada en España con ese nombre, siempre llevado a cabo por una empresa y no por una persona física.

Para consultar los requisitos para el registro de dominios en otros países podemos **visitar las páginas de sus NIC** correspondientes, donde siempre estará todo bien explicado y con la información más actualizada.

Registrar un dominio

En este caso, **los NIC de los distintos países son los que dicen cómo realizar esta labor, y los que realizan el registro**. De todos modos debemos apoyarnos en las mismas empresas que nos servían para el registro de dominios genéricos. También nos pueden ayudar **nuestros proveedores de hosting**, que conocerán bien las normas de registro de los países donde trabajan.

Con respecto al registro de los **.es**, que es el que conocemos, puedes registrarlos también en por medio de nuestro [servicio de registro de dominios](#) y haremos de intermediarios para registrarlo a través de ES-NIC.

Para los dominios **.es**, el coste del registro viene a costar 72,12 € (IVA incluido). Esta tasa permite mantener el dominio durante 1 año natural (es decir, sólo hasta que se termina el año en curso). Posteriormente hay que pagar 48,08 € por año (IVA incluido) para seguir manteniendo el nombre.

Donde conseguir dominios gratis

Es difícil conseguir un dominio gratuito, además, sería un regalo envenenado, ya que la empresa que lo registra gratuitamente se reserva todos los derechos sobre los dominios, como pueden ser la propiedad, el decidir qué uso se le da o la colocación de publicidad, muchas veces abusiva.

Hace tiempo, las empresas que ofrecían estos dominios gratis eran:

- [Namezero](#).
- [DomainZero](#), aunque sólo para ciudadanos de Estados Unidos.

En la actualidad, estas ofertas se han acabado o son de pago (Hay que enterarse de el estado de la oferta en las páginas señaladas, pero no esperéis mucho).

Sin embargo, hay otra opción para conseguir dominios gratuitos muy interesante. Se trata de **DOT.TK**, un dominio asignado a una isla "perdida" que ofrece, para particulares, la posibilidad de registrar gratuitamente nombres de dominio con la terminación .tk.

Para los que desean dominios gratuitos, insistir en que el registro de un dominio generalmente tiene un coste, por lo que no suele ser muy habitual que los regalen. También está la posibilidad de conseguir un subdominio, que no tiene, en teoría, coste de setup, razón por la cuál son regalados muy a menudo.

Nota: Un subdominio es un "dominio dentro de un dominio". Por ejemplo, usuarios.desarrolloweb.com sería un subdominio del dominio principal desarrolloweb.com. www.ciberpais.elpais.es es un subdominio del dominio principal elpais.es.

Podemos encontrar muchas ofertas de subdominios en nuestro directorio, en la categoría [recursos gratis > redirecciones](#).

Esto es todo, esperamos que os haya sido útil toda esta información.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Los 7 dominios nuevos

Los dominios son una manera de identificar un ordenador en Internet, un servidor donde al que se accede, para leer una página web, entrar en un chat, etc.

Existen unas terminaciones para cada dominio que indican el ámbito del servidor donde se está accediendo. Los dos tipos de ámbito más comunes son el regional, que indica el país del servidor, y el genérico, que lo que indica es el tipo de información que se va a encontrar.

La comisión que se encarga de gestionar la administración de dominios ICANN -Internet Corporation for Assigned Names and Numbers- ha aprobado la creación de 7 nuevos dominios genéricos durante última reunión del siglo XX.

Estos dominios de primer nivel se suman a los actuales, de los que tenemos la [descripción y procedimiento de registro](#) en otro reportaje.

La fuente desde donde la hemos recogido esta completa información es Nominalia.es

Veamos cuáles son estos nuevos dominios:

Descripción de los nuevos dominios

.biz

Este dominio es abierto (en el sentido que no hacen falta requisitos específicos previos para el registro, del tipo .com, .net y .org) y el de significado más genérico, por lo que es el que recibirá, probablemente, un volumen de registros mayor. Es la abreviación anglosajona, en pronunciación figurada de business.

.info

Este dominio es también abierto (sin requisitos específicos para registrar) pero el significado parece, a priori, un poco menos genérico que .biz o .com. De todas formas, será también un dominio de gran volumen de registros, seguramente por encima del millón de nombres ya en el primer año.

.name

Este dominio es (bastante) abierto, pero para un uso específico, de carácter personal. Está reservado a los individuos, que podrán reservar su nombre con la estructura MiNombre.MiApellido.name, ej: (pedro.garcia.name). El tercer nivel, correspondiente al nombre de pila, es exclusivo del titular, pero el segundo nivel, correspondiente a los apellidos, es compartido con todos los que ostenten dicho apellido. En este caso, Pedro no podría impedir el registro de miguel.garcia.name, por ejemplo. Son 14 millones de nombres en 5 años los previstos por Global Name Registry.

.pro

Este dominio es para un uso específico reservado a profesionales de determinadas categorías, agrupados en subdominios: inicialmente serán .med.pro (médicos), .law.pro (abogados) y .cpa.pro (auditores; cpa significa chartered public accountant). En el tercer nivel estará el nombre del profesional en cuestión que deberá acreditar su pertenencia al colegio u organización profesional correspondiente.

.coop

Este dominio está reservado a las cooperativas. Un dominio claramente restringido en sus políticas de registro (hace falta demostrar la cualidad de cooperativa a través las organizaciones locales correspondientes). El nombre de dominio debe ser necesariamente el de la cooperativa. Este dominio tendrá un periodo de lanzamiento/test de seis meses, con procesos aún más restringidos, por lo que no va a tener un gran volumen de entradas en el primer momento.

.aero

Este dominio, también de uso restringido, es para la industria de los servicios aéreos: compañías aéreas;

compañías aeronáuticas; aeropuertos y servicios aéreos. El volumen esperado por el registro es de entre 100.000 y 300.000 nombres en cuatro años.

.museum

Este dominio es de uso restringido para la comunidad de museos. La posibilidad de registrar en el segundo nivel (mnac.museum) o en el tercero (mnac.bcn.museum o mnac.sp.museum) según clasificaciones geográficas todavía está por definir definitivamente. Los promotores esperan unos 50.000 nombres en este dominio.

Enlaces de interés:

ICANN: <http://www.icann.org/>

[Descripción de dominios y su registro](#)

Informe de **Miguel Angel Alvarez**

Director desarrolloweb.com

Mail: eugim@desarrolloweb.com

Secretos de los buscadores

Los buscadores y los directorios son herramientas programadas por seres humanos especializadas en buscar información en la Red por medio de sus robots de búsqueda, también llamados Spider o Arañas. De esta manera cualquier internauta que solicite información mediante la introducción de palabras claves o frases cortas en un buscador, obtendrá inmediatamente una lista de páginas web relacionadas con la palabra clave escogida.

Buscadores y directorios

Hay que aclarar la diferencia entre buscador y directorio, son conceptos que normalmente se confunden.

Un directorio como Yahoo depende de humanos o editores que indexan cada URL manualmente, siguiendo su propio criterio de valoración y colocando cada URL en la categoría y subcategoría adecuadas. Por ello el alta en un directorio es un proceso lento y puede llegar a tardar hasta 2 meses.

Una buena página con un buen contenido tiene más posibilidades de ser indexada en un directorio con un ranking alto que una página pobre, sin información, que incluso puede ser rechazada. Los directorios suelen ser muy selectivos a la hora de indexar nuevas páginas. Prefieren calidad y no cantidad de páginas web.

Ejemplos de grandes directorios son Yahoo, Terra y DMoz.

Envía a los directorios solamente tu página principal, redactando un buen título y descripción pensado de antemano a conciencia, utilizando las palabras claves más relevantes y repitiéndolas con frecuencia. El texto que introduzcas al hacer el alta será el que el editor utilizará a la hora de catalogar tu página en la categoría correspondiente.

Los buscadores como Altavista o Excite utilizan un proceso totalmente diferente para indexar las páginas web. Crean sus listados automáticamente por medio de los motores de búsqueda.

El Spider visita tu web, la lee y sigue todos los links a otras subpáginas que vaya encontrando. El Spider volverá a visitar tu página en un mes o dos y si has hecho cambios importantes en la página, el buscador reconoce estos cambios que pueden afectar el posicionamiento de tu web.

Las Metatags son también un elemento a tener en cuenta en un buscador a la hora de extraer la información de la página

(no todos los buscadores soportan Metatags)

En un buscador se pueden destacar tres partes: El Spider que visita las URL, el índice o catálogo que contiene una copia de cada url que ha sido encontrada por el Spider y el software.

Este es el programa que filtra la información grabada en el índice o catálogo, extrae las búsquedas y posiciona cada URL siguiendo un orden o criterio.

Todos los buscadores contienen estas tres partes básicas pero ninguno de ellos funciona igual. Por eso una búsqueda por una palabra clave en particular en Altavista no produce los mismos resultados que la

misma palabra clave en Excite. Estos buscadores siguen criterios diferentes a la hora de indexar las páginas web. Si tu página web tiene predeterminado el criterio que sigue el buscador para indexar las URLs puedes alcanzar unos de los primeros puestos en el buscador. Es fundamental tener una buena organización HTML y saber cómo funcionan los buscadores en general para lograr tener un buen posicionamiento en el buscador.

Factores que afectan al ranking de tu URL

- Utilizando tu propio dominio tendrás muchas más posibilidades de obtener una buena posición y tu página estará más valorada. Si tienes posibilidades no dudes en invertir en tu propio dominio.

- El título es el primer criterio que valora un buscador. El título es imprescindible y recuerda colocar tu palabra clave más relevante en él.

- Tus palabras claves deberían ser frases cortas. Pon tus palabras claves en el título y descripción con frecuencia.

- Utiliza Metatags para que tu página sea correctamente indexada por los buscadores. Las MetaTags comunes son el título, descripción y palabras claves. Una buena combinación de texto en el título y descripción y la elaboración de Metatags es muy recomendable para mejorar el posicionamiento en buscadores que soportan Meta Tags

- No recomendamos la utilización de Frames, pero si lo haces deberás incluir los Metatags en la página que distribuye los frames porque será la única información disponible para indexar. Puedes incluir en el NOFRAMES un párrafo descriptivo del contenido del sitio, así como links a las páginas interiores para facilitar el recorrido del robot.

- Utiliza la MetaTag Robots.txt para dar instrucciones al Spider que recorra los links encontrados en tu Sitio Web e indexe todas las demás páginas. Asegúrate que no hay ningún enlace muerto en la página.

```
<META name="robots" content="All">
```

- Utiliza la Metatag Revisit para indicar al Spider cuándo debe volver a visitar tu página web e indexar los cambios hechos de manera que tu página web esté siempre actualizada y activa en el buscador.

```
<META NAME="revisit" CONTENT="15 days">  
( también 30 days)
```

- Otra Tag útil es el ALT text que va asociado a las imágenes o gráficos, este texto forma parte de la página, así que utiliza tus palabras claves más relevantes en este texto.

```
ALT="myrasoft ofrece software de promocion web"
```

- Incrementa la popularidad de tu página web mediante la colocación de links a otras páginas. La popularidad se mide por el número de enlaces a una determinada página.

Puedes medir la popularidad de tu página y la de tu competencia gratuitamente en:
<http://www.linkpopularity.com>.

La mayoría de los buscadores analizan cuantos links hay hacia tu web, valoran tu popularidad e incrementan el ranking. Además cuantos más links tengas mayores son las posibilidades de que tu página sea visitada por los Spiders. Puedes negociar links de calidad con otras páginas web que estén relacionadas con el tema de tu sitio web y que contengan palabras claves semejantes, a la larga generarán un tráfico importante de nuevos visitantes y clientes potenciales.

- Haz un Mapa de tu Sitio Web con links a todas las páginas interiores. Puedes enviar esta página a los buscadores, esta es una buena táctica para ser localizado en los buscadores. Un ejemplo sería:

```
http://www.myrasoft.com/websiteindexsp.htm
```

- No hagas Spam. Spam es enviar tu página una y otra vez al mismo buscador diariamente o semanalmente. Los buscadores tienen limitaciones y penalizan a las páginas que hacen Spam, borrándolas de su índice. Altavista e Infoseek solamente aceptan una página al día por dominio. Yahoo solamente la página principal. Puedes enviar diferentes URLs en días consecutivos, envía tan sólo las más importantes. Otro tipo de Spam es utilizar texto invisible en letra pequeña del mismo color que el fondo de la página, con la misma palabra clave una y otra vez. Los buscadores penalizan este tipo de diseño, obtendrás un ranking bajo o tu página no será indexada. No utilices palabras claves que no estén relacionadas con el contenido de tu página y no las repitas en exceso. Puedes repetir las una o dos veces, pero intercalándolas.

- Envía tu página una vez al mes o siempre que hayas hecho cambios importantes. Una vez que has enviado tu URL hay que ser paciente, los grandes buscadores indexarán tu página en un promedio de dos a ocho semanas. No todos envían un email de confirmación de que tu página ha sido añadida satisfactoriamente, deberás dejar pasar un tiempo y hacer una búsqueda por dominio o por palabra clave. Una aproximación del tiempo que tardan los grandes buscadores que tienen enlaces en castellano en indexar las URLs:

- Altavista, es el más rápido de 2-3 días a dos semanas.
- Excite de 2-3 semanas
- Lycos de 4 a 6 semanas
- Terra es prácticamente inmediato, de 2 a tres días

La mayoría de los buscadores de lengua castellana son directorios. Terra es un ejemplo. Sigue las recomendaciones descritas para los directorios cuando vayas a dar de alta tu URL. Asegúrate que eliges la categoría adecuada al hacer el alta, intenta ser lo más preciso posible. Parece una tontería pero hay muchas páginas que finalmente no se indexan o que no son indexadas correctamente porque no siguieron las instrucciones adecuadamente al realizar el alta, cometiendo errores.

Hay muchos buscadores que están incorporando una opción geográfica. Si a la hora de dar de alta tu Sitio Web eliges tu región el alta será mucho más rápida de validar y obtendrás mejores resultados. Te recuerdo que el contenido del texto de tu web es fundamental.

El término Portal no es lo mismo que un buscador o directorio, estos términos se confunden porque la mayoría de los portales tienen incorporado un buscador. Los Portales incorporan información adicional como noticias, foros, chats...etc.

Espero que esta información te sirva de ayuda en tu estrategia de marketing en Internet.

Informe de **Myra Palacios**
Directora de Myrasoft
Mail: myra@myrasoft.com
URL: <http://www.myrasoft.com>

Registro en buscadores

Una vez tienes realizada tu web el objetivo es que sea visitada. Existen muchas maneras de conseguir esto, una de las más fáciles e inmediatas es el registro en buscadores.

Para conseguir que tu web sea incluida en un buscador debes rellenar un formulario con los datos de la web que deseas registrar. Cada buscador tiene un formulario específico y a menudo el proceso de registro es diferente.

Para encontrar el formulario lo más fácil es que entres en la página principal del buscador y busques un enlace que ponga **Añadir página, Add URL, nueva dirección**, o algo parecido. Ese enlace te llevara al formulario de registro o, en su defecto, a la página donde explica cómo has de registrarte en ese buscador en concreto.

Aun así, existen unos procedimientos básicos de registro que suelen ser repetirse en los buscadores:

- **Para los índices**, los buscadores que tienen categorías y podemos encontrar páginas clasificadas dependiendo de su temática. Se suele navegar a la categoría en la que se desea incluir el web. Allí se busca el botón "añadir página"(o algo parecido) y se encuentra el formulario para el registro. Índices son [Yahoo!](#) o [Terra](#)
- **Para los motores de búsqueda**, que son los buscadores que no tienen porque mantener un índice y que tienen robots que constantemente recorren Internet en busca de nuevas páginas para incluirlas en el buscador. Estos buscadores suelen tener un formulario accesible desde la página inicial, con el enlace correspondiente. No hay que navegar las categorías para acceder al formulario. Motores de búsqueda típicos son [Altavista](#) o [Sol.es](#).

Salta a la vista que registrarte en varios buscadores puede ser una tarea un poco costosa y otro tanto pesada, pues tienes que ir buscador a buscador introduciendo una y otra vez los mismos datos. Afortunadamente existen herramientas multiregistro. Estas herramientas nos permiten registrar nuestra página en distintos buscadores introduciendo una sola vez los datos de la página web.

Existen también dos tipos de herramientas multiregistro:

- **Aplicaciones windows**, como cualquier otro programa, pero que su objetivo es registrar en buscadores. Este tipo de herramientas suelen tener que comprarse, puedes encontrarlas shareware en www.tucows.com por ejemplo, pero suelen estar limitadas en su uso.
- **Herramientas online**, estas nos permiten desde Internet, y por lo general gratuitamente, registrarnos en varios buscadores. No suelen ser tan potentes como las anteriores, pero si más útiles y accesibles por ser gratuitas. En el [buscador de desarrolloweb.com, en la sección de promoción](#) puedes encontrar alguna de estas herramientas. Nosotros estamos preparando una [herramienta multiregistro online](#), que ya se encuentra operativa y creciendo en el numero de buscadores contemplados.

En desarrolloweb.com disponemos de una [sección dedicada a la promoción de páginas web](#). Amplia en muchísimo la información recogida en este informe y recoge unas herramientas para facilitarte la labor de que tu web sea más vistada.

Informe de **Miguel Angel Alvarez**
 Director desarrolloweb.com
 Mail: eugim@desarrolloweb.com

Contacto con navegante

Bien sabido es que una de las tareas más importantes y laboriosas del ciclo de vida de una página web es su mantenimiento, para conservar la web, su tráfico y su importancia con el tiempo. Pero mantener una web no es sólo publicar nuevos contenidos y actualizar los anteriores, entre estas tareas se encuentra una que es fundamental: proporcionar mecanismos para que los visitantes se puedan poner en contacto contigo, y, por supuesto, contestar a cada uno de los mensajes que recibes.

Uno de los errores graves en una web es que no se brinde la oportunidad de contactar con el creador o responsable de contenidos, pero es aun peor que los mensajes no se respondan pues crea una gran sensación de vacío

Existen muchas formas de comunicarnos con el navegante, vamos a ver los mecanismos que están más a nuestro alcance para que a ninguna de nuestras webs les falte la posibilidad de comunicarse con sus visitantes.

Hemos ordenado los siguientes puntos por orden de facilidad en su uso, así los primeros son más asequibles y los siguientes más difíciles de conseguir, aunque a menudo mejores.

Correo electrónico

Lo que nunca debe faltar en una página, por su sencillez y utilidad, por que es la forma más adecuada en muchos casos... Siempre tenemos que incluir una dirección de correo, fácilmente localizable, para que los visitantes puedan comunicarse. Es facilísimo hacer un enlace con una dirección de correo:

Se ha de poner un enlace convencional, pero su atributo **HREF va direccionado** a una dirección de **correo** con la palabra **mailto:**.

```
<A HREF="mailto:eugim@desarrolloweb.com">Escríbeme</A>
```

Quedaría así: [Escríbeme](mailto:eugim@desarrolloweb.com)

Poner un enlace a una dirección de correo en un editor de HTML, para los que trabajáis con herramientas de edicion, también es parecido a como lo hacéis para los enlaces normales, pero con esas diferencias.

Formulario de contacto

Podemos utilizar un formulario para comunicarnos. En el formulario nuestro visitante puede introducir sus datos y la consulta o sugerencia que desea realizar, para, pulsando luego un botón, enviar por correo electrónico a la persona de contacto todos los datos.

Las ventajas de un formulario con respecto a una dirección de correo pueden ser:

- Ayudar al visitante a componer el mensaje, y así incentivar su comunicación.
- Forzar a que el visitante introduzca cierta información que te pueda resultar importante.
- Ofrecer más mecanismos de comunicacion, cada uno elegirá el que más le convenga.

Veamos brevemente ahora cómo construir un formulario en una página web.

Un formulario se coloca entre las etiquetas <form> y </form>. A esta etiqueta le tenemos que incluir varios atributos, estos son:

- ACTION="mailto:eugim@desarrolloweb.com" Para indicarle a qué dirección de correo enviar los resultados del formulario.
- METHOD="post" Para que lo envíe por método post, esencial para enviarlo por e-mail.
- ENCTYPE="text/plain" Por que lo que vamos a enviar es texto.

En resumen, la etiqueta del formulario quedaría así:

```
<FORM ACTION="mailto:eugim@desarrolloweb.com"
      METHOD="post"
      ENCTYPE="text/plain">
<!-- CAMPOS DEL FORMULARIO -->
</FORM>
```

Ahora veamos cómo colocar campos en el formulario, por lo menos los más fáciles:

Campos de texto
Se utiliza la etiqueta <INPUT> de esta manera: <INPUT TYPE="text" NAME="nombre_del_campo" SIZE=10> El atributo TYPE indica que es un campo de texto. NAME es el nombre del campo, tiene que describir lo que hay dentro. SIZE te permite ajustar el tamaño del campo. La etiqueta NO tiene cierre con </INPUT>
Areas de texto
Se utiliza la etiqueta <TEXTAREA> de esta manera: <TEXTAREA NAME="nombre_del_campo" COLS="20" ROWS="4"> Texto inicial dentro del textarea< BR></TEXTAREA> El atributo NAME es el nombre del campo. COLS indica el número de columnas del área de texto y ROWS en de filas.
Botón de envío
Se utiliza la etiqueta <INPUT> de esta manera: <INPUT TYPE="submit" VALUE="Envíalo YA!"> El atributo TYPE indica que es un botón submit (de envío). VALUE indica lo que va escrito dentro del botón. La etiqueta NO tiene cierre con </INPUT>, NO hace falta darle un nombre con NAME

No te olvides colocar antes de </FORM> el botón de envío y ya tendrás el formulario listo para que tus navegantes te manden sus ruegos y preguntas.

[Puedes ver un ejemplo de formulario aquí.](#)

Referencia: Tenemos una serie de [capítulos en el manual de HTML que tratan sobre la creación de formularios](#) de manera detalladísima.

Tenemos también una serie de [consejos para hacer formularios más rápidos, fáciles y agradables.](#)

Lista de correo

Una lista de correo es una dirección de e-mail a la que si enviamos un correo nos lo hace llegar a todos los integrantes de una lista de direcciones.

Con un ejemplo estará más claro. Si todos los integrantes de una lista, cuando quieren conversar entre ellos, en vez de mandar un correo electrónico a todos ellos lo mandan a un servidor y este se encarga de que le llegue el mensaje a todos los de la lista, este servidor era una lista de correo ;-)

Las listas de correo pueden ser muy útiles para que los usuarios que llegen a las páginas se apunten y nosotros, como administradores del sitio web mandaremos correos electrónicos periódicamente a las lista para mantenerlos informados de noticias, cambios en el web y todo aquello que queramos enviarles.

Para tener una lista de correo en tu página puedes contratar el servicio con un proveedor, pero es mucho más fácil y económico usar una lista de correo de las que regalan en varios servidores. Para

montarla en tu web, simplemente has de seguir las instrucciones que te brinde el proveedor. Puedes [conocer varios enlaces a listas de correo para páginas web en nuestro buscador](#).

Libro de visitas

Ahora veremos algún mecanismo adicional para comunicarse con el cliente, pero en estos casos no se trata tanto de una comunicación entre ellos y tu, sino más es una forma de comunicación de todos para todos.

En el caso del libro de visitas está bien claro, es una herramienta donde los visitantes pueden dejar los mensajes que deseen para que estos queden reflejados en la web y así, no sólo los lees tu, sino que también los pueden leer todos los demás usuarios de tus páginas. Esto le da agilidad a la página, dinamismo y hace que los visitantes se sientan integrados en el proyecto, colaboren y vuelvan para ver sus "huellas". Las ventajas son muchas.

En la mayoría de los casos no dispondremos de la tecnología ni conocimientos para implementar esta tecnología, pero existen en el mercado varios libros de visitas personalizables e integrables dentro de webs sencillas, es decir, tu no tienes que hacer nada, ellos te proporcionan todos los recursos y explicaciones para montar un libro de visitas en tu web. Aquí puedes ver varios enlaces a estos sitios:

- Melody Soft: <http://www.melodysoft.com/> (En castellano)
- Cambia.net: <http://libros.cambia.net> (En castellano)
- [Otros enlaces a libros de visitas en nuestro buscador](#)

Forums de discusión

En algún caso podemos pasar a una opción más avanzada de lo que nos ofrecía libro de visitas. Estos son los Forums de discusión, donde la gente puede opinar sobre temas y otros navegantes contestarles, etc. Es muy útil e interesante. No cabe ya destacar sus múltiples ventajas y hits que recibiremos si se tratan temas de interés.

Si no tenemos la infraestructura suficiente para montar nosotros el forum, podemos utilizar los servicios, muchas veces gratuitos, de otros servidores.

Aquí se pueden seguir varios enlaces que nos llevarán a soluciones para implementar estos forums en tu página.

- Melody Soft: <http://www.melodysoft.com/> (En castellano)
 - BoardHots: <http://www.boardhost.com/>
 - [Otros enlaces a foros en nuestro buscador](#)

Conclusión

Hemos visto un mostón de ideas para incluir en nuestras páginas, todas ellas con el objetivo de establecer contacto con el visitante, aunque al final se nos ha ido el tema del planteamiento inicial y hemos incluido mucho más que meras herramientas de marketing. Porque no olvides que todo esto es para hacerte conocer y hacer que los visitantes te recuerden con alegría y a menudo.

Permíteme por último ofrecerte un enlace, corresponde con la [sección de elementos para incluir en páginas web](#) de nuestro buscador, donde están todos los enlaces enumerados antes y donde irán apareciendo nuevos.

Si quieres, puedes descargarte el texto de esta ayuda técnica y los ejemplos.  [contacto.zip](#)
6Kb

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Caracteres especiales

Una página web se ha de ver en países distintos, que usan conjuntos de caracteres distintos. El lenguaje HTML nos ofrece un mecanismo por el que podemos estar seguros que una serie de caracteres raros se van a ver bien en todos los ordenadores del mundo, independientemente de su juego de caracteres.

Este conjunto son los caracteres especiales. Cuando queremos poner uno de estos caracteres en una página, debemos sustituirlo por su código.

Por ejemplo, la "á" (a minúscula acentuada) se escribe "´" de modo que la palabra página se escribiría en una página HTML de este modo: p&acute;gina

Caracteres especiales básicos

En realidad estos caracteres se usan en HTML para no confundir un principio o final de etiqueta, unas comillas o un & con su correspondiente carácter.

<	<	>	>
&	&	"	"

Caracteres especiales del HTML 2.0

Á	Á	À	À
É	É	È	È
Í	Í	Ì	Ì
Ó	Ó	Ò	Ò
Ú	Ú	Ù	Ù
á	á	à	à
é	é	è	è
í	í	ì	ì
ó	ó	ò	ò
ú	ú	ù	ù
Ä	Ä	Â	Â
Ë	Ë	Ê	Ê
Ï	Ï	Î	Î
Ö	Ö	Ô	Ô
Ü	Ü	Û	Û
ä	ä	â	â
ë	ë	ê	ê
ï	ï	î	î
ö	ö	ô	ô
ü	ü	û	û
Ã	Ã	å	å
Ñ	Ñ	Å	Å
Õ	Õ	Ç	Ç
ã	ã	ç	ç
ñ	ñ	Ý	Ý
õ	õ	ý	ý
Ø	Ø	ÿ	ÿ
ø	ø	Þ	Þ
Ð	Ð	þ	þ

ð	ð	Æ	Æ
ß	ß	æ	æ

Caracteres especiales del HTML 3.2

¼	¼	 	
½	½	¡	¡
¾	¾	£	£
©	©	¥	¥
®	®	§	§
ª	ª	¤	¤
²	²	¦	¦
³	³	«	«
¹	¹	¬	¬
¯	—	­	–
µ	µ	º	º
¶	¶	´	´
·	·	¨	¨
°	°	±	±
¸	ç	»	»
¿	¿		

Otros caracteres especiales

×	×	¢	¢
÷	÷	€	€
“	“	™	™
”	”	‰	‰
Œ	œ	ƒ	ƒ
‡	‡	†	†

Para descargar la lista de caracteres especiales:  [caracteres especiales.zip](#) 2Kb

Informe de **Miguel Angel Alvarez**
 Director [desarrolloweb.com](#)
 Mail: eugim@desarrolloweb.com

Usabilidad para PDAs

Una guía para el correcto diseño de sitios webs accesibles desde PDAs. (Personal Digital Assitants)

1. Ser consciente de las limitaciones de los PDAs.

Los ordenadores de bolsillo tienen una pantalla más reducida, menos memoria y menos velocidad de proceso que los ordenadores convencionales.

2. Definir cuidadosamente la estructura del site.

Con las secciones más importantes y enlaces a las mismas desde todas las páginas. La primera página ya ha de mostrar información útil para el usuario, evitando páginas de bienvenida o de selección de idioma.

3. Evitar el uso de tablas.

En la mayoría de los casos basta con dividir el texto con saltos de línea y párrafos. El procesado de las tablas ralentiza la velocidad de carga de la página. Si se utilizan, especificar las dimensiones en porcentajes y sin sobrepasar los 150 pixels de ancho.

4. No utilizar marcos (frames).

Si ya de por sí se ha de intentar evitar su uso en el Internet convencional, con más razón en el Internet móvil: la mayoría de los navegadores no los soportan y restan mucho espacio en la pantalla.

5. Publicar contenidos concisos.

Por la limitación de la memoria y de la pantalla, escoger sólo la información más importante y esencial.

6. Organizar la información cuidadosamente.

Minimizando la longitud del texto (máximo 3 pantallas de longitud) y escogiendo una distribución óptima de los links de navegación (enlaces a otras secciones). Si el texto debe ser extenso, incluir enlaces a distintas partes del mismo para mejorar la navegación.

7. Optimizar los gráficos.

- **En dimensión:** las pantallas de dispositivos Palm OS tienen un tamaño de 150x150, y los Windows CE/Pocket PC de unos 240x320.
- **En tamaño:** escogiendo el número de colores visibles en el PDA. (16 y 8 bits de color; 16, 4 y 2 escalas de grises).

8. Incluir texto alternativo en todas las imágenes.

Con el tag *alt* ofrecemos información a los usuarios que han deshabilitado la carga de imágenes en su navegador.

9. Añadir el tag *handheldfriendly* al comienzo de todas las páginas.

AvantGo sabrá que la página está optimizada para PDAs.
<meta name="HandheldFriendly" content="true"> .

10. Utilizar convenientemente el *caching* de las páginas.

Para aumentar la velocidad de carga de las páginas. No guardar en caché las páginas que se actualizan todos los días (página de noticias), y guardar en cache las páginas que se actualizan raramente (página de créditos).

Informe de **José Julio Ruíz**
Director de PiensaEnPalm.com
Mail: josejrp@pdainfo.net
URL: <http://www.pdainfo.net>

Mp3 vs. Servidores gratuitos

En Internet hay muchas formas por las que los archivos mp3 se han ido difundiendo: software como Napster, Gnutella, Win Mp3 Locator... también a través de buscadores de mp3 como audiofind.com o audiogalaxy.com, pero el mayor medio de difusión es -siempre lo ha sido- las páginas personales de los cibernavegantes.

La red se ha plagado de páginas personales que contienen mp3, y ésto ha generado un gran conflicto con los servidores, al ser ellos en parte "responsables" por alojar archivos de este tipo.

Qué hacen los servidores gratuitos

Al principio uno de los únicos servidores gratuitos en donde se encontraban los mp3 era Geocities.com, pero ahora casi no quedan empresas de servicios en Internet que no ofrezcan hosting gratuito, por lo que uno se puede encontrar con millones y millones de archivos de audio por toda la red. Ésto ha despertado a los servidores como Xoom.com, Demasiado.com, Tripod.com... a hacer un boicot a los uploaders.

Entonces por ejemplo cuando en nuestro servidor FTP tratamos de uploadear un archivo con extensión mp3 a una cuenta de Xoom no salta un ventana diciéndonos que ese tipo de archivos no se permiten en su servidor. U otro caso peor es el de Geocities, que nos permite subirlo, pero a la hora de haberlo hecho nos anulan la cuenta, entonces el que no lo sabe tal vez uploadea 10 temas en mp3 y en una hora se los borran. No es que esté del lado de los uploaders, pero podrían avisar, no?

Uploaders son usuarios de Internet que suben contenidos a los servidores, en este caso canciones MP3.

Qué hacen los uploaders

Pero los uploaders no se quedan atrás e idearon dos formas de pasar por alto estos problemas: el primer método fue comprimiendo el archivo a formato zip, rar o hqx (con el WinZip), cambiándole entonces la extensión de mp3 a esas extensiones, "engañando" a los servidores.

El otro método, tal vez más difícil, es renombrando el archivo a otra extensión (por ejemplo class, bin, pl...), produciendo el mismo efecto que el método anterior.

Conclusión

Pero la guerra continua, pues algunos servidores realizan inspecciones mensuales en las que detectan archivos comprimidos que poseen mp3 o archivos que pueden ser renombrados a mp3. La vulnerabilidad de los servidores cada vez es más poca, pero igual los uploaders siempre seguiran protagonizando su papel de pícaros usuarios de la red a los que no hay barrera que los detenga.

Informe de **Guido Lonetti**
Webmaster de El Marciano
Mail: guidolonetti@ciudad.com.ar
URL: <http://www.elmarciano.com.ar>

Valorar una web

El objetivo de este taller de HTML consiste en colocar un pequeño formulario para que las personas que visitan nuestro web puedan valorarlo rápidamente. Se trata de un ejemplo de recurso sencillísimo que se puede obtener con sólo utilizar un poco de HTML. La sencillez es belleza y compatibilidad con los distintos navegadores, así que veamos sin más el efecto que pretendemos conseguir.

Como se puede ver a la derecha, el formulario nos propone que valoremos la página y al lado tenemos un botón para mandar la dicha valoración al webmaster del sitio.

Valora esta web

Nota: este ejemplo sólo funciona si el visitante tiene configurado el correo electrónico en su ordenador.

La valoración se enviará por correo electrónico a la persona que se indique en el formulario, tal como veremos más adelante. Si un usuario no tiene correo electrónico configurado en su ordenador con el Outlook Express, Netscape Composer, Eudora o similar el mensaje no se podrá enviar.

Puede darse el caso de que el mensaje se cree, pero no se enviará si no tenemos email... así que probablemente lo podamos encontrar en la bandeja de salida del programa de correo.

Para crear un sistema que no funcione a través del correo electrónico del visitante y conseguir así que incluso los que no tienen correo puedan valorarte, necesitaremos utilizar algún recurso avanzado como CGI, ASP o PHP, pero eso es otro tema.

No pretendo explicar en este artículo los formularios en HTML, ya que en DesarrolloWeb.com hay descripciones suficientes sobre su funcionamiento. Para el que no conozca los formularios recomiendo la lectura [varios artículos en el manual de HTML](#).

El código de este ejemplo es tan sencillo como este:

```
<form action="mailto:xxx@tudominio.es" method="post" enctype="text/plain">
  Valora esta web<br>
  <select name="Valoracion">
    <option>1</option>
    <option>2</option>
```

```

<option>3</option>
<option>4</option>
<option>5</option>
<option>6</option>
<option>7</option>
<option>8</option>
<option>9</option>
<option>10</option>
</select>
<input type="submit" value="Enviar">
</form>

```

Se puede copiar y pegar en la página que lo desees. La única línea que habría que modificar es la de la etiqueta <FORM>, donde tenemos que cambiar la dirección de email del atributo action por la dirección donde queramos que llegue el correo con la valoración.

Si quisiésemos que el correo le llegase a yo@midominio.com pondríamos así nuestro atributo action:

```
action="mailto:yo@midominio.com"
```

Eso es todo. Espero que os haya parecido interesante y práctico el recurso.

Informe de **José Manuel Cerrato Bartolomé**

Mail: jomaceba@wanadoo.es

URL: <http://www.jmcweb.org>

Frames sin bordes

Este taller de HTML explica cómo realizar una declaración de frames sin bordes. Para los lectores que no conozcan lo que son los frames o como se definen sería necesario que estudiaran el [manual de HTML](#), por lo menos los [capítulos dedicados a frames](#).

Hemos visto que hay muchos atributos que sirven para eliminar los bordes de los marcos, tal vez demasiados y haya quedado poco claro cual sería la forma exacta de eliminar todos los frames de una vez. Resulta que no se hace igual en todos los navegadores, aunque incluyendo los atributos adecuados para cada navegador estaremos seguros que los bordes no se verán nunca.

En Netscape, simplemente necesitamos especificar el atributo border="0" en el primer frameset. Esta opción también funciona en las versiones más modernas de Internet Explorer.

En Internet Explorer, debemos especificar dos atributos también en el primer frameset. frameborder="0" y framespacing="0".

Si colocamos los tres atributos a la vez en el primer frameset estaremos seguros que no hay bordes, por lo menos en los navegadores más habituales.

La etiqueta frameset con los tres atributos quedaría así:

```
<frameset cols="90,*" border="0" frameborder="0" y framespacing="0">
```

Ejemplo práctico

Vamos a escribir por completo una declaración de frames sin bordes, para que quede todo suficientemente claro.

```

<html>
<head>
<title>Definición de Frames</title>
</head>
<frameset cols="200,*" border="0" frameborder="0" y framespacing="0">
<frameset rows="170,*">
<frame src="pagina1.html">
<frame src="pagina2.html">
</frameset>
<frame src="pagina3.html">
</frameset>
</html>

```

Se puede ver la [página de frames sin bordes](#) en este enlace.

Tabla con las esquinas redondeadas

En este taller de HTML vamos a crear una tabla con las esquinas redondeadas, que nos podrá servir para destacar alguna información en el texto o crear una barra de enlaces lateral un poco más vistosa. A la derecha aparece una tabla como la que pretendemos conseguir.

El ejemplo no resulta nada complejo. Simplemente se crea una tabla normal, en la que colocamos en cada una de sus esquinas un gráfico que hace la forma redondeada. Los gráficos que utilizamos en esta ocasión tienen una parte de color y otra transparente. La parte de color es la que dibuja el borde redondeado y la parte transparente deja ver el color de fondo que hayamos colocado en la tabla. Las imágenes se pueden ver a continuación. Para guardarlas utiliza el botón derecho del ratón encima de la imagen y selecciona la opción que pone "Guardar imagen como..." o algo parecido. También podrás [descargar las imágenes y el ejemplo completo en un archivo comprimido](#).



En nuestro ejemplo hemos creado imágenes que tienen la parte no transparente de color blanco, que corresponde con el color de fondo de la página donde queremos colocar la tabla. Si queremos colocar una tabla como esta sobre un fondo distinto al blanco deberíamos crear unos gráficos que tengan el mismo color que el fondo, en lugar de blanco.

Lo bueno de que el otro color utilizado en la imagen sea transparente es que la tabla que creamos puede tener el color de fondo que se desee. Esta otra tabla -a la derecha- se crea con las mismas imágenes del ejemplo y, como se puede ver, tiene otro color de fondo que la anterior.

Creación de la tabla

Ahora vamos a estudiar el código HTML que hace falta para crear esta tabla con esquinas redondeadas. Probablemente con otro código HTML más simple también se podría construir, pero hemos preferido añadirle un pequeño exceso de atributos y etiquetas que servirá para estar seguros de que se puede visualizar correctamente en todos los navegadores.

La tabla que utilizamos contiene varias celdas dispuestas en tres filas y tres columnas. En las celdas de las esquinas es donde colocamos las imágenes que hacen que los bordes aparezcan redondeados. En el resto de celdas de la tabla que forman el borde, para asegurarnos de que tienen el tamaño correcto, colocamos imágenes de un píxel transparente con sus correspondientes atributos de anchura y altura modificados a lo que necesitamos. En la celda del centro es donde colocamos el cuerpo de la tabla, con todo el texto que queremos que vaya dentro, sus imágenes, etc.

```
<table width=300 cellspacing=0 cellpadding=0
      bgcolor="#333399" border=0>
  <tr>
    <td width=11></td>
    <td width=278></td>
    <td width=11 align=right></td>
  </tr>
  <tr>
    <td></td>
    <td><font color="#ffffff" face="verdana,arial,Helvetica" size=2>
      <b>Tabla guay</b>
    <br>
    <br>
  </td>
  <td></td>
  </tr>
</table>
```

Este es el texto que quieras ponerle a la tabla. Puedes poner tanto texto como desees, que la tabla se hará lo suficientemente grande como para que quepa todo.

```

        </font></td>
    <td></td>
  </tr>
  <tr>
    <td width=11></td>
    <td width=278></td>
  <td width=11 align=right></td>
  </tr>
</table>

```

Si alguno desea utilizar este código para crear sus propias tablas únicamente debería modificar unos pocos datos:

- Texto del cuerpo de la tabla.
- En caso de que se desee modificar el ancho de la tabla
 - + Tamaño de la tabla. Atributo width de la etiqueta <table>
 - + Tamaño de los píxeles transparentes, en la primera y última fila. Atributo width de las etiquetas de los pixels transparentes.

Los anchos de los pixels transparentes (en la primera y última fila de la tabla) tienen que ser el ancho de la tabla menos el ancho de las dos imágenes que aparecen en las esquinas. En nuestro código, como el ancho de la tabla es de 300 píxel y el ancho de las dos imágenes de los bordes es de 11 píxel, el ancho de la imagen de píxel transparente será $300 - 11 \times 2 = 300 - 22 = 278$

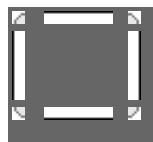
Para acabar, recordamos que se pueden descargar las imágenes, así como el código fuente de la tabla con esquinas redondeadas, en un [archivo comprimido](#).

Informe de **Miguel Angel Alvarez**
 Director desarrolloweb.com
 Mail: eugim@desarrolloweb.com

Tabla con las esquinas redondeadas, tipo 2

Vamos a ver como realizar con HTML una tabla con las esquinas redondeadas y con un pequeño marco de un píxel. Es un ejemplo de tabla con los bordes redondeados como cualquier otro, de hecho, ya hemos visto un ejemplo sobre este asunto en nuestro anterior artículo [Tabla con esquinas redondeadas](#). Como siempre, lo mejor para darse cuenta de lo que pretendemos construir es verlo en un ejemplo y al lado de estas mismas líneas podemos verlo.

En este caso utilizaremos las siguientes imágenes, que podemos guardar pulsando sobre ellas con el botón derecho del ratón y seleccionando "Guardar imagen como...". También podemos [descargar todo el código y las imágenes en un archivo comprimido](#).



Cuerpo de la tabla

Aquí podremos escribir lo que deseemos, que la tabla crecerá lo suficiente para contener todo el texto que coloquemos, incluso imágenes.

Espero que parezca un diseño interesante, aunque seguro que los hay mejores...

Creación de la tabla

Vamos a ver el código HTML necesario para crear este ejemplo. Hemos de tener en cuenta que el código se podría haber creado de varias maneras, aunque nosotros presentamos la que consideramos más versátil.

Hemos construido la tabla con las correspondientes etiquetas de tablas de HTML. Como consideración cabe señalar que las etiquetas de las imágenes que se colocan dentro de las celdas tienen que estar pegadas a la etiqueta </TD>, que se utiliza para cerrar la tabla. Si no es así puede que nuestro ejemplo quede descuadrado.

El tamaño de la tabla en anchura se puede definir perfectamente en el atributo width de la etiqueta <TABLE>. La altura será la suficiente para que quepan todos los contenidos de la tabla.

Este es el código en cuestión:

```
<TABLE WIDTH=300 BORDER=0 CELLPADDING=0 CELLSPACING=0>
  <TR>
    <TD>
      <IMG SRC="images/tablita_01.gif" WIDTH=6 HEIGHT=6></TD>
      <TD background="images/tablita_02.gif">
        <IMG SRC="espacio.gif" WIDTH=1 HEIGHT=6></TD>
      <TD>
        <IMG SRC="images/tablita_03.gif" WIDTH=6 HEIGHT=6></TD>
    </TR>
    <TR>
      <TD background="images/tablita_04.gif">
        <IMG SRC="espacio.gif" WIDTH=6 HEIGHT=1></TD>
        <TD bgcolor=E8E8E8 valign=top>
          Cuerpo esto es el Cuerpo esto es el Cuerpo esto es el Cuerpo...
        </TD>
        <TD background="images/tablita_06.gif">
          <IMG SRC="espacio.gif" WIDTH=6 HEIGHT=1></TD>
        </TR>
    <TR>
      <TD>
        <IMG SRC="images/tablita_07.gif" WIDTH=6 HEIGHT=6></TD>
        <TD align=center background="images/tablita_08.gif">
          <IMG SRC="espacio.gif" WIDTH=1 HEIGHT=6></TD>
        <TD>
          <IMG SRC="images/tablita_09.gif" WIDTH=6 HEIGHT=6></TD>
        </TR>
  </TABLE>
```

No hay mucho que explicar... simplemente que [recojas las imágenes descargando el archivo comprimido](#) (a notar que han sido colocadas en un directorio llamado images y que si no están allí no funcionará el ejemplo) y que la imagen que se llama espacio.gif es simplemente un pixel transparente.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Algunos consejos para webmasters

Algunos consejos para webmasters distribuidos por el autor del artículo en la [lista de correo de ayuda](#) de DesarrolloWeb.com.

1.- JAMAS programen directo sobre la PC, diagramen todo lo que vayan a hacer, en terminos informaticos se llama "algoritmizar", y a nivel empresarial se llama "trabajo de mesa", ese es el verdadero corazon del trabajo....nunca dejen que los problemas les sorprendan a mitad del trabajo, eso desmoraliza al programador perdiendo asi su interes en el trabajo y conllevando a una mala realizacion del proyecto.

2.- TODOS los nombres que usen traten de que sean lo mas relacionado con el contenido de los que trata, eso mejora a la hora de analizar los enlaces y/o la estructuración del sitio; de igual manera pasa con las variables a la hora de paginas activas en cualquiera de los lenguajes, y en el caso especifico de variables traten de definir las antes de usarlas, o sea, forcen la definicion de las variables cada vez que puedan asi a la hora de analizar los errores sera mas comodo.

3.- El diseño de un sitio es todo un arte, JAMAS se lo tomen a la ligera. NO se debe programar NUNCA para un determinado tipo de explorador, eso demerita a los clientes cuando entran. La SOBRIEDAD en la sitios es algo digno y no un mal diseño, los diseños BARROCOS se deben usar solo para sitios donde tienen la VERDADERA necesidad de mostrar grandes volúmenes de informaciones al mismo tiempo, si su sitio no admite mas contenido NO lo presione. Como norma se establece no mas de tres colores y no mas de tres tipografías en cada pagina, lo mas recomendable es usar un "sello único" a la hora de diseñar, o sea, que el cliente se acomode perfectamente de un solo vistazo a su sitio.

En el plano personal les digo que el fondo blanco es muy refrescante.

PD.: Mas adelante les dare algunos consejos mas, saludos y suerte;

Informe de **Alberto García Gómez**
Adminitrador Nodo Atenas
Mail: alberto@yumuri.mtz.sld.cu

Cómo colocar un foro en tu página web

Para disponer de unos foros en tu página web tienes dos opciones principales. La primera es utilizar un servicio de foros de otra página web. La segunda es instalar unos scripts y una base de datos en nuestro alojamiento que hagan en trabajo desde nuestro propio dominio. Esta segunda opción es más interesante pero, lógicamente, requerirá más recursos a tener en nuestro sitio web y más conocimientos para ponerlo en marcha.

Utilizar un servicio de otro sitio web

Hay muchos sitios web que ofrecen foros para colocar en nuestras páginas. Estos foros suelen ser ligeramente configurables e integrables en nuestro sitio rápidamente y con poco esfuerzo. Como contrapartida, la empresa que ofrece el servicio suele incluir publicidad en las páginas del foro y la integración con nuestro sitio será menor. Otra desventaja, más obvia pero no menos importante es que las opciones para mejorar el servicio están fuera de nuestro alcance, ya que la administración del servidor de foros corre por cuenta de la empresa que ofrece el servicio.

En este sentido, en DesarrolloWeb.com podemos encontrar muchos enlaces de sitios que ofrecen foros para nuestra web. Los veremos todos en nuestra sección de [recursos gratis > elementos para webs > foros](#).

Instalar una aplicación de foros

Si deseamos ofrecer unos foros a lo grande, con muchas opciones de configuración y de adaptación a nuestro sitio, con posibilidad de administración y, en definitiva, con un control total de la aplicación, lo mejor es que instalemos una aplicación de servidor.

Es necesario que contemos con un alojamiento que permita algún tipo de programación en el servidor y alguna base de datos. Como posibles tecnologías de servidor podemos señalar CGI, JSP, ASP o PHP, si bien las dos últimas cuentan con muchas más posibilidades y desarrollos gratuitos y listos para usar. Las bases de datos podrán ser variadas, pero dependerán de la tecnología con la que estemos trabajando.

Referencia: como ya hemos dicho, instalar un foro ya creado requerirá que sepamos un poquito sobre algún lenguaje del servidor. Si no sabemos nada de ellos, probablemente lo que digamos a continuación y la propia instalación del foro, no tendrá mucho sentido. Si es así, queremos apuntar los enlaces más interesantes de DesarrolloWeb para encontrar documentación sobre las tecnologías de servidor.

[ASP a fondo](#)

[PHP a fondo](#)

[Categoría JSP de nuestro directorio](#)

[Categoría CGI de nuestro directorio](#)

Cuando descarguemos alguna aplicación de estas, generalmente, nos encontraremos con un archivo comprimido Zip que contendrá las páginas web que conforman el foro y la base de datos que se utiliza en el desarrollo. Las páginas habrá que subirlas por FTP y la base de datos colocarla en el servidor, aunque la realización de esta segunda tarea dependerá de qué base de datos sea la que deseamos subir.

Generalmente también nos encontraremos con algún o algunos archivos de configuración del foro, que tendremos que editar para que funcione todo en nuestro servidor y con las características que deseemos. Pero todo esto seguramente vendrá explicado perfectamente en las instrucciones que seguro acompañan al foro en el archivo comprimido.

Para encontrar foros ya listos para usar podemos acceder a páginas como [Hotscripts](#) (buscar por Discussion Boards dentro de los scripts de la tecnología que estemos tratando) o <http://sourceforge.net/> (podemos hacer una simple búsqueda por la palabra forum).

Informe de **Manuel Alamar**
Mail: manuelalamar@guiarte.com

El posicionamiento en los buscadores de Internet

Todas las empresas con cierta entidad, hoy en día disponen de un sitio web, pero muchas de ellas se plantean si esta presencia en la Red ha sido una inversión o un mero gasto de representación.

Para que nuestra presencia en Internet sea rentable es necesaria la promoción de nuestro sitio, y una de las herramientas más rentables para dar a conocer nuestro site y generar visitas son los buscadores.

Aparecer en los principales buscadores internacionales, nacionales y en los específicos de nuestro sector es principal, pero no lo es menos aparecer en un lugar destacado según ciertas palabras clave que definan nuestro negocio en la mente de nuestras audiencias. Puesto la mayoría de los internautas se conforman con los primeros resultados proporcionados por su buscador favorito.

Y para aparecer en un lugar destacado en estas útiles herramientas de búsqueda debemos conocer su funcionamiento interno. Los buscadores se dividen en dos grandes grupos: los índices y los motores de búsqueda.

Referencia: en nuestro [manual de promoción de páginas web](#) podemos encontrar gran parte de la información de este artículo, relatada con mayor detalle y más calmadamente.

Los índices dividen la información en un árbol temático de categorías y subcategorías. Aquí el ejemplo paradigmático sería Yahoo!, que nos presenta una serie de grandes categorías temáticas entre las que encontramos la subcategoría "Economía y Negocios", dentro de ella "Empresas", y esta a su vez contiene entre otras "productos y servicios para empresas" y así sucesivamente hasta ir acotando la amplitud de la categoría de sitios web, ya que no se nos permitirá proponer la inclusión de nuestro sitio web en una categoría demasiado amplia. En los índices lo esencial es encontrar la rama ideal de este árbol temático en la cual ubicar nuestro sitio web, y digo nuestro sitio, puesto que en los índices sólo es posible incluir una página (normalmente la principal) a su directorio, aunque a menudo es posible incluirlo en dos o tres categorías. Para encontrar esta categoría ideal en la cual debería estar nuestro web, la estrategia a seguir es puramente marketiniana; ponerse en la piel de nuestro público y pensar en qué categoría nos buscará. Para ello, podemos ayudarnos de un estudio de mercado, y como no, del sentido común y de la observación de en qué categoría se encuentran ubicados nuestros principales competidores. Pero cuidado, quizá ellos no lo hayan echo tan bien y no se encuentren en la categoría ideal. Lógicamente dependiendo de la amplitud de nuestros productos o servicios, será más obvio o más difícil hallar esta categoría ideal. No obstante tras nuestra petición de alta existe un proceso de revisión humano e incluso podemos proponer una nueva categoría si no nos encontramos debidamente definidos por ninguna de las existentes.

Y sobre este proceso de revisión humana es sobre el que quiero hablar a continuación, puesto que es este el segundo factor que más diferencia a los índices de Internet de los motores de búsqueda. Cuando proponemos el alta de nuestro sitio en el índice de turno, se nos pide toda una serie de datos, como: Título de la página, URL, Definición, Ubicación geográfica, persona de contacto y correo electrónico... Y finalmente nuestro site es revisado por un surfer (un especialista en catalogar recursos) del índice que considera si nuestro site cumple con los estándares de calidad requeridos y si está bien clasificado en la categoría elegida por nosotros.

Vemos que esto es lo único que conoce el índice de nuestro sitio web; los datos suministrados en el formulario de petición de alta en el buscador. Por lo que debemos ser extremadamente cuidadosos en la definición que enviamos de nuestro site.

El caso de los motores de búsqueda es bien distinto. Podemos tomar como ejemplo a Google, y veremos que la única información que proporcionamos a un motor es la dirección URL (por ejemplo: www.miempresa.com) y quizá una dirección de correo electrónico. El resto del proceso se realiza de forma automática, ya que nuestra petición de alta en el buscador entrará en la cola de trabajo de un programa de software llamado spider (araña) que visitará la página que hemos dado de alta y a partir de ella todas las que se encuentren enlazadas y así sucesivamente. Simultáneamente nuestras páginas serán indexadas utilizando complejos algoritmos, para ser devueltas como resultado cuando un internauta utilizando el buscador, introduzca un término que se encuentre en alguna de ellas y haga una petición de extracción de información de su ingente base de datos. Vemos de esta forma que nuestro web puede aparecer en algún motor de búsqueda por la simple razón de que otra página de un tercero que está incluida en el buscador enlaza a ella en Internet.

Así, en los motores de búsqueda, para obtener una notable posición, lo esencial es el código de nuestras páginas, algo que era verdaderamente indiferente en el caso de los índices.

Teóricamente con sólo dar de alta nuestra página principal el buscador indexará todas las páginas que

cuelgan de ella, pero habitualmente nos encontraremos con problemas derivados de la ventaja que se concede a las altas de pago frente a las gratuitas; el primero es el tiempo a esperar para que nuestro sitio sea introducido en la base de datos del motor de búsqueda, que puede variar entre varias semanas a varios meses según el motor en cuestión. E incluso a menudo, tras este dilatado periodo de tiempo, no seremos indexados en sus bases de datos. Y esto, en los motores que aún admiten el alta gratuita.

Como recomendación, si nuestro tiempo y energías son limitadas deberemos optar por el alta de pago en algunos buscadores.

Si conocemos las interacciones entre los distintos buscadores de Internet, descubriremos que la inclusión en alguno de ellos puede suponer la sucesiva inclusión en otros que a menudo son más "duros" con las admisiones.

Una vez que conseguimos que nuestro web aparezca en los buscadores, nuestro trabajo no habrá hecho más que empezar, ya que lo realmente valioso es aparecer en los primeros lugares por aquellas palabras clave que nuestros públicos utilizan para buscar nuestra categoría de productos, y esta sí que es una verdadera guerra, puesto que en esa lucha estamos frente a nuestros principales competidores, que también batallarán por mejorar la posición de sus páginas frente a las nuestras y las de otros competidores. Desde luego, que la complejidad dependerá de la popularidad de las palabras clave por las que queramos aparecer de forma destacada en el buscador.

Realmente cada buscador valora de distinta forma el código de nuestras páginas para ubicarla en una u otra posición de su ranking, así por ejemplo Google valora especialmente cuantas y que tipo de páginas apuntan hacia las nuestras, aplicando una lógica bastante humana, según la cual si muchos y especialmente importantes hablan de uno, es que uno es importante. Otros motores como AltaVista valoran los Meta Tags (unas líneas de código que informan al motor acerca del contenido de nuestras páginas), etc.

En general, los buscadores se fijan en la frecuencia o densidad y ubicación con la que aparecen ciertos términos en nuestras páginas, para ubicarlas en un lugar superior de sus resultados frente a otras páginas, en las cuales la frecuencia y ubicación de este término que el navegante ha introducido en la caja de búsqueda del motor aparece.

Así, una palabra que está presente en nuestra misma dirección de Internet (www.palabra.com) indica un elevado nivel de coincidencia si es el término buscado por el internauta. Después es especialmente valorada esta palabra, en el título del documento, en el primer párrafo más que en el segundo... Si está en mayúscula es más valorado, al igual que si está en negrita... Si aparece dos veces en una frase más que si aparece una, etc.

A estas alturas, seguro que a más de un lector se le ha pasado por la cabeza, la idea de llenar de términos clave la página para que aparezca en las primeras posiciones del buscador, pero lamentablemente esto ya está contemplado por estas herramientas que si encuentran demasiadas palabras repetidas o un texto de tamaño muy pequeño o con el mismo color que el fondo, etc, penalizarán nuestras páginas o incluso las eliminarán de la base de datos por tratarse de técnicas de spam (técnicas de promoción ilícitas)

Informe de **Javier Maldonado**
Director de Marketing
Mail: jmaldonado@marketalia.com
URL: <http://www.marketalia.com>

Creación de un menú desplegable en Dreamweaver

Muchas veces, nuestras páginas tienen tantos contenidos que si quisiéramos que aparecieran todos estos en un determinado lugar de nuestra página, en un menú, este abarcaría casi la totalidad del espacio que tenemos para nuestra web. En estos casos es muy útil insertar uno o varios de estos menús desplegables, apareciendo por encima del propio contenido de la página para mostrar todas sus partes y desapareciendo posteriormente.

A lo largo de este artículo vamos a desarrollar la forma de crear uno de estos menús con DreamWeaver de una manera genérica para que cada uno lo amplíe a su gusto.

En primer lugar, tenemos que saber que este menú está construido casi en su totalidad por capas, a las cuales les atribuiremos unos comportamientos específicos para que aparezcan y desaparezcan a nuestro gusto.

Otra cosa que debemos tener en cuenta es la condición de capa absoluta o capa relativa, ya que a las capas absolutas se les tiene que dar unas coordenadas de posicionamiento a raíz de la esquina superior izquierda de nuestra web, coordenadas que no nos sirven para nada en el caso de que nuestra página

tenga los contenidos centrados, ya que la posición en la que va a aparecer esta, dependerá de la configuración del monitor desde el que se visualice dicha página.



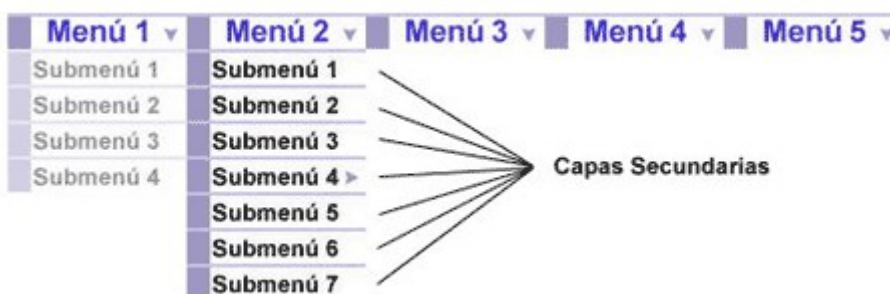
Menú Principal

El primer paso que tenemos que dar es desarrollar mentalmente un esquema del menú que queremos realizar, o mejor, sobre papel. Nosotros nos hemos decantado por hacer un menú principal compuesto por 5 partes en posición horizontal, para lo cual crearemos una tabla con 5 celdas y les daremos a cada una de ellas un determinado el tamaño de píxel, en nuestro caso 100px - 20px. A continuación haremos clic en el icono de capa y dibujaremos una en cualquier lado, seguidamente arrastraremos el símbolo de capa (que determina el lugar donde el programa introducirá la línea de código HTML, que por defecto creará dentro de la etiqueta "Body") dentro una de las celdas de la tabla que hemos generado y a continuación modificaremos en la ventana propiedades, los campos "Iz" (izquierda) y "Sup" (superior) dejándolos en blanco, al hacer esto el programa engancha la capa en la esquina superior izquierda del recipiente en el que se encuentra, en este caso, la celda en la que hemos introducido la capa. Después daremos un valor a los campos de "An" (ancho) y "Al" (alto) En el ejemplo que estamos creando serán 100px y 20px respectivamente, este paso lo tenemos que repetir para cada una de las 5 celdas (en nuestro caso), de las que se compone nuestro menú principal.

Una vez terminado con este paso procederemos a crear otras capas dentro de las que ya hemos establecido, a estas capas tendremos que darles unos valores de tamaño dependiendo de las distintas partes que queramos introducir dependientes de cada menú (nosotros daremos 100px - 80px en el primer desplegable, 100px - 140px en el segundo, 100px - 100px en el tercero, 100px - 80px en el cuarto y 100px - 140px en el quinto). Para colocar una capa dentro de otra podemos hacerlo: 1º, arrastrando como hemos hecho anteriormente esta nueva capa dentro de la anterior, o 2º, presionando la tecla F2 nos aparecerá la ventana "capas" donde podremos ver un esquema de las capas que tenemos en nuestra web, cogiendo una de las capas que aparecen y arrastrándola encima de otra mientras que presionamos la tecla "ctrl", introduciremos la capa arrastrada dentro de la que hayamos seleccionado.

Estas subcapas que hemos creado, que dependen de las principales, por defecto tienen la propiedad "default" que deja la capa visible en todo momento y nos viene bien para trabajar, pero antes de darles un comportamiento a estas, cuando tengamos terminado por completo la estructura de nuestro menú, deberemos cambiar este tributo "default" de las capas secundarias (las que se encuentran dentro de las 5 capas principales) por "hidden", que las hace invisibles, dándonos la posibilidad de trabajar con los comportamientos haciéndolas aparecer y desaparecer a nuestro gusto.

Dentro de estas capas secundarias introduciremos tablas con el número de celdas que hayamos calculado con el tamaño de cada capa. Una vez hecho esto en todas las capas deberíamos colocar las imágenes o las palabras de nuestro menú para poder configurar los comportamientos, ya que si no colocamos nada dentro de las capas, estas no se verán cuando aparecen o desaparecen por ser transparentes. Otra solución es darles un color de fondo.



El último paso para terminar nuestro menú será dar a cada capa un comportamiento. Para eso necesitamos la ventana comportamiento que, si no la tenemos ya a la vista, presionando la tecla F3 aparecerá. Para poder jugar con los comportamientos de "mostrar u ocultar capa" debemos trabajar con un vínculo (estos vínculos serán las imágenes que hemos colocado dentro de las celdas y en su defecto las palabras, para hacer este vínculo introduciremos en el espacio reservado para los vínculos de la ventana propiedades el símbolo "#" creando así un vínculo en blanco, a continuación presionaremos el vínculo creado (en la parte inferior de la página principal aparecerá el símbolo de vínculo "<a>" en negrita) después iremos a la ventana comportamiento y presionaremos el símbolo "+" y pincharemos en "Mostrar - Ocultar capas" y nos aparecerá una ventana con un listado de todas las capas que tenemos creadas, lo que tenemos que hacer es presionar la capa que se tendría que desplegar cuando pasásemos el ratón por encima del vínculo que estamos modificando y apretar el botón "Mostrar" y las demás capas pincharlas y ocultarlas de la misma forma que hemos hecho antes pero en este caso apretando el botón

de "Ocultar".

Este paso lo repetiremos para cada una de los cinco vínculos principales de los que se nos desplegarán los correspondientes submenús.

Informe de **David Marín**
Mail: davidmarin@mundofree.com

Tabla mejorada con imágenes para barra de navegación

En este taller de HTML vamos a ver como una pequeña imagen puede resultar muy vistosa para construir una barra de navegación para nuestro sitio web.

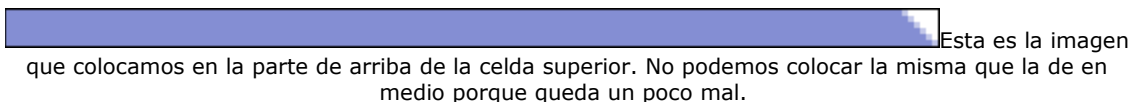
Vamos a construir una tabla como la que se puede ver en la parte de la derecha, donde podremos observar la utilización de imágenes para camuflar el hecho que las celdas son siempre rectangulares. Las imágenes aplican un leve biselado y eliminan una esquina, con lo que las tablas mejoran sensiblemente su apariencia.

Portada
Introducción a ello
Todos los asuntos
Que desees
Más epígrafes
Acabando los enlaces
Contáctenos

Nota: Un efecto como este o parecido se puede conseguir de muchas maneras, así que nos tenemos que tomar este taller como tan sólo una idea de las posibilidades y el modo de construir las tablas.

Las imágenes

Creo que viendo las imágenes que hemos colocado en la tabla se comprenderá un poco la idea sobre la que hemos trabajado. Las imágenes están ampliadas para que se pueda observar mejor sus líneas. Se pueden crear con cualquier editor gráfico del que dispongamos.



Consideraciones para crear la tabla

Vamos a colocar cada elemento en la tabla en una celda independiente. En la primera celda colocaremos la imagen destinada para la parte de arriba, en la segunda el texto del primer enlace, luego la imagen que colocamos en medio de cada celda de texto, seguida por otra celda con el texto del siguiente enlace, luego otra vez la imagen, luego texto, etc.

La tabla se tiene que crear de modo que no quede separación entre celdas ni márgenes, pues si la hubiera no parecería que las celdas de la imagen y las del texto forman un mismo bloque y aparecería deslabazada. Los atributos `cellspacing` y `cellpadding` quedarían a cero.

Además, las etiquetas `<TD>` y las de las imágenes, ``, tienen que estar en el código sin espacios entre medias, pues si no fuera así tampoco conseguiríamos que las celdas quedasen pegadas unas a otras.

Por lo demás, decir que las celdas de texto les hemos aplicado estilos utilizando CSS (Hojas de estilo en cascada), que son mucho más cómodos y nos permiten definir una única vez el estilo para todas las celdas en lugar de repetir las etiquetas y atributos HTML para cada una.

Código de la tabla

Colocamos el código de toda la página en lugar de solamente la tabla para que se puedan ver las etiquetas para colocar los estilos CSS, que aparecen en la cabecera.

`<html>`

```

<head>
<title>Tabla enlaces guay</title>
<style>
.celda {background-color:#848ED3;font-size:8pt;font-family:verdana,arial;color:#ffffff;font-weight:bold;padding-left:3px;padding-bottom:2px;}
</style>
</head>

<body>

<table cellspacing="0" cellpadding="0" border="0">
<tr>
<td></td>
</tr>
<tr>
<td class="celda">Portada</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td class="celda">Introducción a ello</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td class="celda">Todos los asuntos</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td class="celda">Que deseese</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td class="celda">Más epígrafes</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td class="celda">Acabando los enlaces</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td class="celda">Contáctenos</td>
</tr>
</table>

</body>
</html>

```

Se puede [ver el ejercicio en una página a parte](#).

También ponemos a vuestra disposición la [descarga del archivo HTML de la tabla y las las imágenes utilizadas en el ejemplo](#).

Informe de **Miguel Angel Alvarez**
 Director desarrolloweb.com
 Mail: eugim@desarrolloweb.com

Parcheando aplicaciones

Bien saben los técnicos de sistemas que hay que estar siempre al tanto de las actualizaciones de los programas de ordenador, ya que incluyen solución a los agujeros de seguridad que se van detectando. Sin los parches somos vulnerables a los virus y programas malignos. Ningún usuario, por muy novato que sea, debe olvidarlo.

Durante el transcurso de un año, podemos encontrar del orden de 10 a 20 actualizaciones críticas en software de uso común que solucionan agujeros de seguridad importantes y que nadie debería dejar de instalar, igual que nadie se olvida sus zapatos para no lesionarse las plantas de los pies al pasear por la calle.

Hay que estar al tanto de las actualizaciones y para ello podemos visitar las páginas web de los desarrolladores del software que utilizamos habitualmente, brindando especial atención al sistema operativo o los programas de Internet. Resultará también muy útil leer cada cierto tiempo publicaciones con noticias tecnológicas o especializadas en seguridad, como Hispasec. <http://www.hispasec.com/>

Parchear el sistema operativo

El caso más clásico es el de Windows, el sistema operativo de la mayoría de los ordenadores personales del mundo y también una de las mayores fuentes de vulnerabilidades. Sólo tomando este sistema operativo en su versión más moderna de la actualidad, Windows XP, podremos encontrar 5 o 6 actualizaciones críticas, aun siendo un producto presentado hace relativamente poco tiempo. Si nos referimos a otros sistemas operativos de Microsoft, como Windows 98, el número de parches críticos que podremos encontrar supera la decena, debido a que este sistema es mucho más antiguo y, por lo tanto, ha habido mucho más tiempo para detectar agujeros de seguridad.

Microsoft por lo menos cuida a sus usuarios y pone muy a la vista sus actualizaciones, críticas o no tanto. Accediendo a la página <http://windowsupdate.microsoft.com> podemos encontrar una aplicación web que revisa nuestro sistema operativo Windows, independientemente del modelo o versión y nos informa de las actualizaciones que debemos introducir. El informe recoge tanto las actualizaciones críticas, que no debemos dejar de instalar, como las actualizaciones importantes, que no se refieren a problemas de seguridad, sino a detalles relevantes sobre el trabajo con el ordenador, como puede ser una adaptación del sistema al Euro.

Aunque no todo son bondades en los sistemas de actualización de Microsoft. Para empezar, el simple hecho de disponer de tantas actualizaciones resulta cuando menos polémico. Sin embargo, la nota más negativa resulta que Microsoft va "jubilando" sus sistemas operativos con el tiempo. Por ejemplo, con la entrada de 2003 ha calificado de obsoletos a sus sistemas MS-DOS, Windows 3.x, Windows 95 y Windows NT 3.5. Windows ME, con tan sólo unos pocos años de vida, se jubilará el 31 de diciembre de 2003. A partir de entonces, Microsoft no ofrecerá más asistencia ni actualizaciones para ellos, lo que puede dejar desprotegidos a miles de usuarios de todo el mundo, que deberán actualizar su sistema operativo para disponer actualizaciones de seguridad.

Windows no es el único sistema operativo donde podemos encontrar vulnerabilidades. En realidad, cualquier sistema construido por el hombre está potencialmente afectado y Linux o Mac OS no son una excepción. Bien es cierto que las vulnerabilidades encontradas son en número menores, pero también es verdad que el ímpetu con el que se buscan los agujeros en Windows es mucho mayor, debido a que algunos informáticos utilizan buena parte de su tiempo y se divierten buscando manchas en el expediente de Microsoft.

Por otro lado, decir que hay menos vulnerabilidades en Linux que en Windows, puede resultar erróneo para la opinión de ciertas empresas, o por lo menos, una verdad a medias. Al cabo del año se realizan muchos informes por consultoras que sitúan por delante, en la dudosa clasificación de agujeros de seguridad, a unos u otros sistemas, dependiendo, lógicamente, de quién encargue el informe. Es un tema al que es preferible no entrar, dado que cada cuál tiene sus impresiones y, como vemos, hay muchas opiniones contrastadas. Lo que sí está claro es que existen muchos menos virus y usuarios con problemas de seguridad en Linux y que la comunidad que desarrolla el sistema está siempre atenta a los agujeros que se puedan encontrar para solucionarlos a las pocas horas de haber sido detectados.

Parchear programas

No sólo los sistemas operativos están afectados por las vulnerabilidades, también los programas, muchos de ellos de uso habitual, pueden afectar nuestros ordenadores, brindando a las personas maliciosas agujeros de seguridad por los que acceder a nuestros equipos y realizar tareas dañinas.

Cuanto más cerca de Internet están los programas o los ordenadores, más problemas de seguridad pueden aparecer. Uno de los ejemplos de programas con serias vulnerabilidades que se detectan periódicamente es Flash, una herramienta para la visualización de contenido dinámico y multimedia en páginas web que tienen instalada el 97% de los ordenadores, según Macromedia, la compañía que desarrolla el producto. Hace poco en Flash fueron encontradas algunas vulnerabilidades para las que es necesario actualizar el visor a la versión más moderna para poder evitarlas. Macromedia no ofrece una

información clara en su página para que los usuarios actualicen sus programas, podríamos decir que esconde sus vulnerabilidades y con ello la posibilidad de que los usuarios las subsanen rápido.

Otros programas para los que se han detectado vulnerabilidades importantes son los llamados P2P (Peer TO Peer) que sirven para compartir, de igual a igual, programas y ficheros multimedia entre los usuarios de Internet. Estos ficheros algunas veces están infectados por virus que utilizan vulnerabilidades para ejecutar código dañino, agujeros que las discográficas siempre reciben con buenos ojos puesto que los programas para compartir información, generalmente música, resultan muy contraproducentes para sus actividades.

El paquete de herramientas Office, también de Microsoft, para el trabajo de oficina presenta también varias vulnerabilidades importantes para las que existen los correspondientes parches. Microsoft informa sobre ellos en la página web <http://office.microsoft.com/ProductUpdates/>

Conclusión

Es normal que, una vez lanzado un producto software, se encuentren vulnerabilidades importantes que no se había tenido en cuenta antes. Debido a que la creación y distribución de nuevas ediciones de los programas resulta muchas veces muy costosa y es casi imposible hacer llegar a todos los usuarios un parche o una nueva versión del programa comprado, debemos estar al tanto de las actualizaciones que se van presentando en el software. Es importante instalar las actualizaciones y, de ese modo, estar protegidos contra posibles agujeros de seguridad.

Las actualizaciones sólo se presentan cuando se ha detectado un problema y puede que, incluso antes de que se distribuya un parche para corregir el error, existan programas o virus que utilicen los agujeros de seguridad para hacernos padecer serios quebraderos de cabeza con nuestros ordenadores.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Desabilitar la barra de imágenes de Internet Explorer

¿Cuántas veces nos hemos quejado o hemos deseado que no apareciese esa incómoda barrita en Internet Explorer (Versión 6) cuando pasamos el puntero por encima de una imagen?. Pues la solución es extremadamente simple...

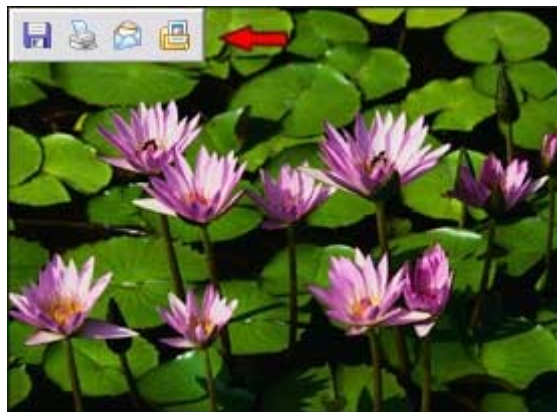


Imagen que muestra la barra emergente.

Tan sólo, lo que tenemos que incluir en la cabecera de nuestra página, es una escueta etiqueta META que automáticamente nos deshabilitará la susodicha opción.

También disponemos de un comportamiento de Dreamweaver que nos ahorrará el tener que escribir dicha META. Pero cuando veáis lo reducida que es, probablemente paséis de descargaros el comportamiento. De todas formas para todos aquellos que deseéis tenerla, la podréis adquirir en la web de [Macromedia Exchange](#).

Esta es la etiqueta META que deberéis incluir entre <HEAD> y </HEAD>

```
<meta http-equiv="imagetoolbar" content="no">
```


Y lo veremos un poquito mejor en un pequeño ejemplo de código HTML, de modo que no quede ningún tipo de duda.

Ej:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//SP">
  <html>
  <head>
    <!-- #BeginEditable "doctype" -->
    <title>::: Mi página web:::</title>
    <!-- #EndEditable -->
    <META NAME="TITLE" CONTENT="Nombre">
    <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
    <meta name="Description" content="descripción de nuestra web.">
    <meta name="Keywords" content="palabras relacionadas con el contenido de nuestra web.">
    <meta name="author" content="Nombre de la empresa.">
    <META HTTP-EQUIV="EXPIRES" CONTENT="Mon, 31 Dec 2054 00:00:01 PST">
    <meta name="reply-to" content="correo electrónico de la empresa">
    <meta name="owner" content="Propietario de la empresa.">
    <META HTTP-EQUIV="VW96.OBJECT TYPE" CONTENT="Tipo de web">
    <META NAME="RATING" CONTENT="General">
    <meta name="robots" content="index, follow">
    <META NAME="REVISIT-AFTER" CONTENT="7 days">
    <meta http-equiv="imagetoolbar" content="no">
  </head>
  <body>

    El contenido de nuestra web.

  </body>
</html>
```

Recordad que no importa donde coloquéis la etiqueta META, siempre y cuando esta se encuentre entre el <HEAD> y </HEAD>.

Informe de **Christian Santalucía**
Mail: hidden@terra.com

Frontpage 2000 para principiantes

Introducción a Frontpage para principiantes

Es relativamente común oír por la Red, o entre colegas diseñadores —cuando se tienen amigos tan extraños como uno mismo—, que las peores páginas de Internet no fueron desarrolladas por un mal webmaster, sino escritas con FrontPage. Y tristemente es muy probable que lleven razón. (¿Triste? De momento retiro el adjetivo; no se puede sentir lástima de un producto fabricado por la mayor compañía de software del mundo.) ¿Por qué es así? Además, tengamos en cuenta que es una herramienta muy, muy extendida; ¿pueden equivocarse cientos de miles, quizás millones de personas? La respuesta a esta pregunta es sí, pueden equivocarse, y la otra cuestión requiere una argumentación un poco más extensa.

FrontPage está programado por los empleados de [Microsoft](#), la madre amantísima de Internet Explorer, y salta a la vista que se esmeraron en que los usuarios de Netscape nos sintiésemos defraudados. A eso se le llama simpatía corporativa. Pero ¿qué más da? Siempre nos quedará el semejante y más neutral y accesible [Namo Web Editor](#). [FrontPage](#) genera lo que podríamos llamar código extraño, una especie de tiras de texto cifrado, cabalístico, que sólo utiliza el programa y seguramente nadie más en el mundo. A eso lo define como código fuente. El problema es que, además, [FrontPage](#) crea archivos y etiquetas basura, que aumentan el peso de nuestros websites y el tiempo de transferencia y publicación.

A pesar de todo esto, es una aplicación que inspira en los usuarios la simpatía de lo conocido. Forma parte de la suite Microsoft Office y su interfaz es muy parecida a la del omnipresente Word y a la del resto de productos de la gama. Por otra parte, sus archivos de ayuda son muy completos. Esto sin duda es una ventaja para el principiante. Sin embargo, el profesional del medio se encontrará con la imposible tarea de construir un palacio con herramientas de plástico, a diferencia de aquél que ejecuta software del prestigio y la calidad contrastada de [Macromedia DreamWeaver](#) o [Adobe GoLive!](#), tan bien integrados, por si fuera poco, con otras aplicaciones comunes y muy potentes, de gran calidad visual y multimedia, como [Flash](#) y [PhotoShop](#).

Pero siendo un producto muy usado, especialmente por aficionados cuyas finanzas no dependen de su talento creativo, se merece al menos una introducción a algunas de sus funciones. No es un manual para un uso avanzado, sino un repaso elemental de las preguntas más frecuentes que los usuarios plantean en los foros de discusión, así como una manera de hacer compatible su código con otros navegadores. Una vez instalado e iniciado el programa —digamos que cuando ya es demasiado tarde para echarse atrás—, nos encontramos con la pantalla principal de modo WYSIWYG; siglas en inglés de "lo que ves es lo que obtienes". La reconocemos porque en la parte inferior de la pantalla, sobre la barra de estado, vemos seleccionada la pestaña "normal"; observemos que hay dos más: "html" y "vista previa". La vista html nos muestra el código fuente, que es lo que interpreta el navegador, y es especialmente importante en un producto de las características de [FrontPage](#), dado que su código debe ser revisado y corregido para adaptarlo a navegadores diversos y para incluir código [javascript](#), por ejemplo. La última pestaña, vista previa, nos muestra el resultado tal y como se verá en Internet Explorer, y además ejecuta una función de depuración de código: si hay errores, nos lo hará saber mediante una ventana emergente.

Barras de herramientas de Frontpage

Existen varias barras de herramientas, y la mejor idea es probar todas ellas, con todas sus funciones, hasta descubrir cuáles son las que realmente resultan útiles de forma frecuente, y aquellas de uso más inusual. Al final es normal que creemos nuestra propia barra, para eliminar del escritorio elementos innecesarios.

Las barras de herramientas más comunes para el usuario principiante son: estándar, con los botones típicos de guardar, abrir, página nueva, deshacer, rehacer, etc.; formato, para aplicar tipografías y párrafos al texto (FrontPage admite la justificación de párrafos, pero solo desde Formato > párrafo > alineación, sin disponer de un botón específico en la barra de formato); tablas, para construir tablas y modificar sus propiedades; imágenes, para insertar gráficos, cambiarles el tamaño y la resolución, crear mapas de imágenes, modificar luz, contraste, etc.; efectos **DHTML**, para aplicar animaciones automáticas o de reacción a las páginas; exploración, para controlar la estructura de un sitio web; e informes, que nos detalla las páginas lentas, anticuadas, etc.

Existen otras barras de herramientas, como por ejemplo de formulario, pero requieren ciertos conocimientos adicionales y, de momento, el usuario novel no las necesita... ni entiende.

Informe de **TC**

Mail: sunjammer8@netscape.net

URL: <http://www.tobcazamian.net>

Creando un web en local con Frontpage

Para empezar, hay que tener claras dos nociones elementales y sencillas.

- Una página web es un único archivo, una estructura de datos e informaciones que instruyen al navegador sobre cómo debe mostrar dicha página.
- **Un sitio web** es una carpeta o grupo de carpetas que contiene a su vez páginas web individuales y, posiblemente, otros archivos como imágenes, animaciones **Flash**, documentos **JavaScript**, películas **QuickTime** o **Real**, etc.

En **FrontPage** debemos crear primero una carpeta web, que es algo así como la versión local, en nuestro disco duro, del sitio web accesible al público y colocado en un servidor. En realidad nunca deben crearse páginas arbitrariamente, en un folder cualquiera, ya que eso dificulta considerablemente la tarea de publicación. La operación es fácil: clic sobre archivo, clic sobre nuevo, clic sobre web... La aplicación nos da la opción de elegir entre algunas plantillas; mi opción predilecta, más libre, es "web de una sola página", de modo que yo me encargo de todo el trabajo posterior, asumiendo un control absoluto.

Se introduce la ubicación y el nombre de la carpeta web, en el campo al efecto, y se acepta. Unos segundos después, ya disponemos de nuestra web local, donde **FrontPage** ubicará todas las páginas y el resto de archivos. Así, al publicar, no tendremos pérdida.

Definiendo la resolución con Frontpage

Un problema fundamental del diseño web es la proyección de nuestras páginas, es decir, cómo pensamos que serán vistas por el usuario. ¿Qué tamaño tiene su pantalla? ¿Es igual que la nuestra, más grande?

Inevitablemente ocurrirá una de las dos cosas, e incluso ambas. Aunque es cierto que los monitores de 800x600 son muy comunes, también lo son los de 17". Por ello, si pensamos en el primero, dejaremos vacío gran parte del espacio del usuario de 17"; y si pensamos en éste, obligaremos al navegante de 15" a utilizar las barras de desplazamiento horizontal para moverse por una página, y eso es muy incómodo.

Referencias: Tenemos algunos artículos que tratan el tema de las resoluciones de pantalla en páginas web con más detenimiento y de manera general a todos los programas:

- [Definiciones de pantalla](#). Cómo tratarlas.
- [Páginas fluidas](#) (las que se ajustan al tamaño de la ventana del explorador) Cuando son apropiadas.

La solución predilecta del diseñador suele consistir en crear tablas con un ancho y alto de tamaño definido porcentualmente, es decir, adaptable a cada monitor; el 100% es una buena opción. El procedimiento es simple: Tablas > Insertar: a continuación aparece una ventana, donde introducimos a voluntad ciertos datos, como grosor del borde, número de filas (horizontales) y columnas (verticales), color o imagen de fondo (la imagen de fondo en las tablas ya es reconocida en [Netscape 7](#)). En esta primera ventana podemos definir el ancho: 100%, esto es, adaptable al monitor. Una vez que aceptemos, ya disponemos de nuestra tabla en la pantalla.

Ahora, haciendo clic con el botón derecho del ratón sobre la misma tabla, seleccionamos Propiedades de tabla en el menú contextual. Ahora sí, definimos también el alto en 100%. Conviene advertir, en todo caso, que este código es interpretado por [Internet Explorer](#), pero no por [Netscape](#) en ninguna de sus versiones. Para hacerlo accesible a todos, podemos cambiar a la vista html y sustituir la tabla (definida por las etiquetas <table> </table>) por este código:

```
<table cellpadding="0" cellspacing="0" border="0" width="100%" height="100%"
style="text-align: left;">
<tbody>
<tr>
<td valign="top">
</td>
</tr>
</tbody>
</table>
```

Este código ha sido generado por [Netscape Composer](#) y será asimilado por [FrontPage](#), con posibilidad de editarse normalmente sin que se pierda su compatibilidad (conviene que, en [FrontPage](#), sigamos estas instrucciones: abrimos Herramientas> Opciones de página> pestaña Código fuente html y seleccionamos "mantener el código fuente html", de manera que [FrontPage](#) no rescriba el código

generado por otras aplicaciones). Ahora, todo lo que componemos dentro de la tabla se adaptará a los diversos monitores, siempre quedando dentro de la pantalla.

Si tan sólo queremos definir el ancho de la tabla, nos limitamos a seguir las primeras instrucciones. En todo caso, ningún usuario se verá obligado a mover el scroll horizontal cuando navegue nuestros sitios. (A excepción de aquellas ocasiones en que incluyamos objetos muy grandes sin posibilidad de reestructuración automática, como por ejemplo las imágenes; si bien sería posible definir su tamaño como porcentaje de la ventana, esta opción tan sólo es útil en gráficos sin degradados, ya que las fotografías quedarían deformadas y pixelizadas).

Informe de **TC**

Mail: sunjammer8@netscape.net

URL: <http://www.tobcazarian.net>

Funciones especiales de Frontpage: Las extensiones de servidor

Conviene precisar que esta aplicación de Microsoft incluye algunas opciones particulares, muy cómodas e interesantes, como por ejemplo la creación de foros totalmente personalizados o la inclusión de contadores. Sin embargo, cuenta con la desventaja de que estas características dependen del servidor donde alojemos nuestras páginas, y si éste no cuenta con las extensiones apropiadas, la mejor idea es desechar todas estas funciones adicionales... porque sencillamente no funcionarán. Si ya conocemos nuestro proveedor de espacio web, gratuito o de pago, podemos consultar sus servicios y ver si cuentan con estas extensiones; normalmente muestran esta información de manera explícita, ya que es una pregunta muy común. Si no disponen de las extensiones, podemos deshabilitarlas todas de manera automática...

Hacemos clic en el menú Herramientas> Opciones de página> pestaña Compatibilidad. Deseleccionamos la casilla "habilitado con las extensiones de servidor de Microsoft FrontPage". Así, sus las funciones asociadas quedan atenuadas e inoperativas en los menús, de manera que no nos ocurrirá algo muy común e irritante: descubrir al final que parte de nuestro trabajo ha resultado en vano.

En esta ventana, además, contamos con otras opciones muy útiles, y con algunas más que podrían sonar a chiste. La mejor idea suele consistir en seguir estas instrucciones:

—En la entrada Exploradores, seleccionar "personalizado". Es de muy mal gusto planear un web pensando tan sólo en un grupo cerrado de usuarios.

—En la entrada Versiones de exploradores, seleccionar "personalizado". En [FrontPage 2000](#) se tratan tan sólo hasta las versiones 4.0, mientras que en la actualidad Internet Explorer avanza por la 6, y Netscape, por la 7.

En el listado de tecnologías cada usuario puede elegir las opciones que le convenzan. Para el caso de [ActiveX](#), por ejemplo, hay que considerar el hecho de que éste no es soportado por Netscape, mientras que existen otras opciones compatibles con ambos navegadores, si lo que buscamos es incluir contenido

multimedia en nuestras webs.

Informe de **TC**

Mail: sunjammer8@netscape.net

URL: <http://www.tobcazamian.net>

Multimedia con FrontPage 2000

Como decía, los controles [ActiveX](#) son particulares de Internet Explorer. Se trata de pequeños códigos visibles en la fuente del documento que indican al navegador ciertas instrucciones para definir el comportamiento de contenido multimedia y dinámico. En general suelen ser entradas complicadas para el usuario novel, y llegan a variar mucho de un control a otro; es decir, quizás tengas los conocimientos suficientes para incluir flash, pero no los apropiados para añadir [RealVideo](#). Si no se conocen, son inútiles. En todo caso, usualmente incluyen la opción de "representación alternativa" para exploradores que no soportan esta tecnología.

Entonces, **¿qué hacer para insertar músicas y contenidos especiales?** Como curiosidad valga decir que la ventana propiedades de página (fácilmente accesible haciendo clic con el botón secundario del ratón sobre la pantalla principal) tiene la opción de música de fondo, pero en general es una mala idea. Ocupa ancho de banda, puede resultar pesada y la mejor alternativa por su escaso peso, la música midi, suena soporífera e insustancial.

Bien, la opción entonces es la inclusión de complementos (Insertar> Avanzadas> Complemento), que generará unas etiquetas `<embed>` `</embed>` interpretadas tanto por Netscape como por Explorer. Las entradas que debemos cumplimentar en la ventana que aparece son los siguientes: "origen de datos", que determina la fuente o archivo de origen; el tamaño en alto y ancho (aunque puede ocultarse el complemento si se desea, seleccionando la casilla Ocultar, que creará la etiqueta "hidden"); y ciertas variables de diseño que pueden ser editadas más tarde, y más cómodamente, desde los controles de la barra de formato.

¿Qué podemos hacer con los complementos o archivos embebidos? Pues en realidad agregar cualquier archivo que un navegador sea capaz de interpretar mediante plugins; desde [Flash](#) de [Macromedia](#) hasta [QuickTime](#) o músicas mp3. Sin embargo, cada archivo específico tiene unas características particulares cuyo código debe ser conocido y editado desde la vista de código fuente. Por ejemplo, las animaciones flash incluyen `menu="0/-1"`, que muestra o no, el menú contextual, etc. Quizás pueda verse de una forma más clara el resultado del código, que deberá editarse manualmente, en la siguiente línea:

```
<embed src="archivo.xxx" width=320 height=200 AUTOSTART=true  
LOOP=true></embed>
```

Desde FrontPage podremos añadir `autostart`, que indicará si el inicio de la reproducción del archivo es automático o corresponde a un evento, y `loop`, que define si el fichero se ejecuta indefinidamente o si tan solo se reproduce una vez y luego se detiene. Existen más variables genéricas y otras específicas de formato; en todo caso, la [página de desarrollo de Netscape](#) incluye información complementaria muy interesante.

Informe de **TC**

Mail: sunjammer8@netscape.net

HTML limpio

Como ya se dijo anteriormente, [FrontPage](#) trabaja también en modo vista de código fuente, que es completamente editable y cuyos resultados se actualizan en tiempo real en la vista WYSIWYG. Sin embargo, el usuario novel, para quien va destinada esta introducción elemental, quizás no conozca dicho lenguaje, y lo que desea hacer es simplemente cortar y pegar códigos que tal vez ha bajado de la Red. **¿Es esto posible?** Desde luego, y de hecho de dos formas distintas. Para una de ellas usaremos el comando Insertar> Avanzadas> Html, que abre una ventana en blanco donde pegar cualquier secuencia. Antes deberemos especificar con el ratón dónde queremos insertar el nuevo código. No obstante, éste se incluirá según está escrito, sin análisis, y su contenido no se mostrará en la vista normal.

Para eliminar esta característica deberemos borrar los comentarios que el software genera automática. Acudimos a la vista html y borramos completamente y sin temor las líneas: `<!--webbot bot="HTMLMarkup" startspan -->`, `endspan -->`, al principio y al final.

La otra forma la usaremos para insertar código en el encabezado del documento, que está delimitado por las etiquetas `<head>`, `</head>`, donde la función insertar html no es efectiva.

Para ello pasamos a la vista de código, donde pegamos cualquier código de la forma clásica, abriendo un menú contextual con el ratón y seleccionando pegar. No obstante, puede haber un problema: si el texto copiado tiene formato, FrontPage lo conservará, haciendo anotaciones de formateado, e inutilizará el script. Puesto que corregirlo manualmente sería un engorro, podemos seguir un pequeño truco: primero pegamos el texto en el [bloc de notas](#), de manera que se pierde el formato, y allí volvemos a copiar, apareciendo limpio de etiquetas basura.

Conclusiones:

Bien, éstas son algunas de las preguntas que comúnmente se pueden leer en los foros de usuarios de [FrontPage](#), y aquí exponemos algunas posibles respuestas. No son todas, ni siquiera tienen que ser infalibles; hay más opciones creativas que el usuario puede descubrir con la práctica, disfrutando así de un trabajo hecho con cierto ingenio.

Informe de **TC**
Mail: sunjammer8@netscape.net
URL: <http://www.tobcazamian.net>

PROMOCIÓN DE PÁGINAS WEB

Introducción a la promoción de webs

Una vez hemos construido una página web tenemos que **hacer que esta sea conocida** por todos los medios que estén a nuestro alcance, para atraer visitas a ella y, cuando menos, sentirnos orgullosos de que esta sea popular en la Red. Para conseguir esto tenemos que promocionarla adecuadamente, de manera que su dirección figure en el mayor número de sitios.

Existen muchas maneras de promocionar una página,

- Podemos enviar correos anunciando su puesta en marcha
- Registrarla en los buscadores
- Incluir la dirección en el membrete del papel de nuestro negocio
- Anunciarla por televisión, por poner varios ejemplos.
- Una vez la web ya lleva un tiempo online podemos hacer que los visitantes vuelvan a ella mediante técnicas adicionales como listas de correo que informen a los usuarios de las novedades.

Este manual pretende dar a conocer las técnicas para conseguir los anteriores objetivos, centrándonos en el más importante: **el registro en buscadores**, que requiere una serie de indicaciones muy específicas para que nuestra página este presente y ocupe los **primeros puestos en los resultados de la búsqueda**.

Pero atención, no es solo nuestro trabajo promocionando una página el que va a atraer las visitas, también debemos mantener la web con buena presencia, que sea rápida y que tenga unos buenos contenidos. En otras palabras, con la misma promoción, una buena página web atrae más visitas que una mala, algo muy lógico si lo piensas.

Este manual está compuesto por varios capítulos. En los primeros vamos a conocer al detalle todos los mecanismos y los trucos relativos a los buscadores. En capítulos más avanzados veremos otras maneras de promocionar la página.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Tipos de buscadores

Existen varios tipos de buscadores en Internet. En primer lugar, podemos distinguirlos por su forma de trabajo, esto es importante, dado que **la manera de registrar una dirección en los buscadores es diferente según el tipo**.

Indices

Son los buscadores que mantienen una **organización de las páginas** incluidas en su base de datos **por categorías**, es decir, tienen un directorio navegable de temas. Dentro de cada directorio podemos encontrar páginas relacionadas con ese tema. Para mantener esta organización, los buscadores tienen unos administradores humanos que se encargan de visitar las páginas y vigilan que todas se encuentren clasificadas en su lugar correcto. Índices típicos son **Yahoo**, **Terra** o **TodoEnlaces**.

Para que una página quede registrada en un índice debemos **mandarles la dirección a los administradores** humanos de ese índice, generalmente

acompañada de una serie de datos que les ayuden a clasificar la página de una forma correcta, como la descripción, temática, título, lenguaje, etc. Además, si queremos que varias páginas de nuestro sitio web estén en el buscador, **deberemos registrarlas todas ellas una a una.**

Motores de búsqueda

Son buscadores que basan su **recolección de páginas en un robot**, denominado araña, que recorre constantemente Internet en busca de páginas nuevas que va **introduciendo en su base de datos automáticamente.** Los motores de búsqueda, no tienen porque tener un índice, aunque cada vez es más habitual que dispongan de uno. Motores de búsqueda típicos son **Altavista** o **Sol.**

Los motores de búsqueda, como se puede haber deducido, no necesitan que les mandemos la dirección de nuestra página para tenerla en su base de datos, puesto que el robot puede haberla encontrado previamente. De todos modos, nosotros podemos **mandarles la dirección si no deseamos esperar a que el robot nos encuentre**, practica muy habitual.

Para clasificar una página, los motores de búsqueda son capaces de leer el contenido de esta y encontrar aquellos datos que permitan su catalogación. Por esto es que cuando registramos una página en un motor de búsqueda generalmente no nos piden información adicional, como ocurría con los índices.

Cuando un robot recorre nuestra página guarda sus datos, y luego se dirige a las distintas páginas que están enlazadas a esta. De este modo, solo hace falta registrar la página inicial de un sitio web, pues el motor de búsqueda se encargará de **recorrer todo el sitio de manera automática.** Adicionalmente, estos motores o arañas, volverán a recorrer las páginas de su base de datos en busca de cambios que se hayan producido en estas, con objetivo de mantener su información lo más actualizada posible.

Multibuscadores

Estos últimos no tienen una base de datos propia, lo que hacen es buscar la **página en unos cuantos motores de búsqueda** e índices y combinar los resultados de la búsqueda en esos buscadores. Como ejemplos de multibuscadores podemos destacar **Meta buscador de desarrolloweb**, en castellano, o **Metacrawler**, en inglés.

Para registrar una dirección de manera que aparezca en un multibuscador debemos mandársela a los algún buscador donde este va a recoger los resultados.

Otras clasificaciones

Aparte de la clasificación anterior de los buscadores, también se pueden distinguir de otras formas.

Por su ámbito: los hay internacionales, nacionales, incluso de regiones más pequeñas, como provincias o ciudades.

Por el tema: existen buscadores genéricos, donde podemos encontrar todo tipo de páginas, y también los hay temáticos, donde solo hay páginas que tratan sobre una temática específica.

A la hora de registrar una página debemos comprobar que pertenece al ámbito o temática del buscador, o de lo contrario es casi seguro que no la aceptarán.

En el siguiente capítulo vamos a ver cómo se ha de registrar una página en cada tipo de buscador.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Cómo registrar una página

Para conseguir que tu web sea incluida en un buscador debes **rellenar un formulario con los datos de la web** que deseas registrar. Cada buscador tiene un formulario específico y a menudo **el proceso de registro es diferente**.

Para encontrar el formulario lo más fácil es que entres en la página principal del buscador y busques un enlace que ponga **Añadir página, Add URL, nueva dirección**, o algo parecido. Ese enlace te llevara al formulario de registro o, en su defecto, a la página donde explica cómo has de registrarte en ese buscador en concreto.

Aun así, existen unos procedimientos básicos de registro que suelen repetirse en los buscadores:

Para los índices

Los buscadores que tienen categorías donde podemos encontrar páginas clasificadas según su temática. Por lo general se ha de **navegar a la categoría** en la que se desea incluir el web. Allí se busca el botón "añadir página"(o algo parecido) y se encuentra el formulario para el registro.

Para los motores de búsqueda

Que son los buscadores que no tienen porque mantener un índice y que tienen robots que constantemente recorren Internet en busca de nuevas páginas para incluirlas en el buscador. Estos buscadores suelen tener un formulario accesible desde la página inicial, con el enlace correspondiente. **No hay que navegar las categorías** para acceder al formulario.

Salta a la vista que registrarte en varios buscadores puede ser una tarea un poco costosa y otro tanto pesada, pues tienes que ir **buscador a buscador introduciendo una y otra vez los mismos datos**. Afortunadamente existen herramientas multiregistro. Estas herramientas nos permiten registrar nuestra página en distintos buscadores introduciendo una sola vez los datos de la página web.

Existen dos tipos de herramientas multiregistro:

Aplicaciones windows, como cualquier otro programa, pero que su objetivo es registrar en buscadores. Este tipo de herramientas suelen estar a la venta, o puedes encontrarlas shareware (en www.tucows.com por ejemplo), pero suelen estar limitadas en su uso.

Herramientas online, estas nos permiten desde Internet, y por lo general gratuitamente, registrarnos en varios buscadores. No suelen ser tan potentes como

las anteriores, pero si más útiles y **accesibles** por ser **gratuitas**. En desarrollo web disponemos de una **herramienta multiregistro online**, vigilada de cerca para que todo funcione bien. Existen otras de estas herramientas en Internet, se pueden ver en la **[sección de promoción de nuestro buscador](#)**.

Por otro lado, hay que destacar que existen una serie de métodos que nos pueden ayudar a que nuestra página se encuentre mejor situada entre los resultados. pero esto es otro tema, que se tratará en el siguiente capítulo.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: **eugim@desarrolloweb.com**

Trucos para el registro

Antes de empezar el proceso de registro en buscadores es muy recomendable, incluso absolutamente necesario, que las páginas **hayan sido revisadas y modificadas, incluyendo todas aquellas ayudas que permitan una mejor clasificación** de las mismas. De este modo, conseguiremos que nuestras páginas sean fácilmente localizables entre los resultados del buscador.

Existen unas reglas y recomendaciones básicas para conseguir este objetivo. Elegir bien el título, descripción o palabras clave son algunos ejemplos. Vamos a ver con detenimiento en este reportaje cada una de estas técnicas. No nos tenemos que olvidar tampoco que el uso excesivo de estas puede ser contraproducente ya que muchos motores de búsqueda interpretan que la **página esta haciendo "trampa"** para situarse entre los primeros puestos de manera injusta. Cuando nuestra página es catalogada como "spam page" o "página tramposa" (en adelante **spam**), es confinada a los últimos puestos en los resultados de manera automática. Hay que utilizar estas técnicas con moderación, como se puede ver, por regla general.

Antes de abordar el tema sería adecuado conocer **varios conceptos que vamos a utilizar**:

Palabras clave

Son palabras sueltas que describen una página web, las que pensamos que van a **escribir los usuarios en los buscadores** para encontrar una página como la nuestra.

En este manual, cuando hablemos de palabras clave nos referimos a **palabras sueltas**, pero en la práctica, tanto en la búsqueda de documentos como en la preparación de nuestra página para ser reconocida por los buscadores eficientemente, se utilizan las frases clave, descritas a continuación.

Frases clave

Son las **frases que van a escribir los usuarios** en los buscadores para encontrar un tipo de página. Si buscan mecánicos de coches de Vallecas, una población de la comunidad de Madrid, podrán escribir algo como mecánicos de vallecas madrid, reparación coche vallecas madrid, taller mecánico vallecas madrid, taller coche vallecas...

Las frases clave, como se ve, deben incluir las **distintas combinaciones de palabras clave** que se supone que van a utilizar los distintos usuarios para

localizar nuestra página. Escogerlas bien es importante y para ello podemos valernos de un buscador. Haciendo pruebas entre distintas combinaciones de frases clave podemos comprobar cuales son las que ofrecen unos resultados mejores, en las que los usuarios, es de suponer, que se fijarán más, rebuscando incluso en páginas sucesivas de resultados.

Es interesante el uso de frases clave por que son utilizadas habitualmente por los usuarios en sus búsquedas. Además, si utilizas frases clave es más posible que alguien te encuentre pues posiblemente no tengas tantos competidores como los que tendrías con una palabra simple.

Descripción

Es una **frase que describe el contenido de la página**. Se corresponde con la descripción que un buscador ofrece de cada página en los resultados de la búsqueda.

Caracteres especiales

Existe una duda generalizada sobre la utilización de los **acentos, mayúsculas, minúsculas** y otros caracteres. Nosotros recomendamos **escribir todas las palabras y frases correctamente, usando los acentos y las mayúsculas donde sea necesario**. Esto es así porque cuando buscamos, en Altavista por ejemplo, una palabra y la buscamos con acento sólo nos devuelve las páginas que contemplan esa palabra con los acentos, mientras que si la buscamos sin acento nos las ofrece todas, las que lo tienen como las que no.

Ahora veamos cómo hacer llegar a un motor de búsqueda todos los conceptos anteriores de palabras y frases clave, así como la descripción.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Dónde colocar las palabras clave

Ahora vamos a ver **dónde situar** las palabras y frases clave, así como la descripción dentro de la página web.

El Título de la página

Lo primero que debemos hacer es escoger un título apropiado, es decir, un título que describa bien la página donde se encuentra y contemple otras pequeñas consideraciones:

- **Tiene que ser muy descriptivo**. Algo como Taller mecánico no es suficiente. Debería indicar también la localización del taller o cualquier información adicional que pudiera enriquecer la descripción, como el nombre del taller o su especialidad.
- Ha de **incluir las palabras clave** que definen nuestro sitio. Además debería contener las palabras más importantes al principio, pues muchos buscadores solo se fijan en las primeras palabras del título.
- La longitud del título también es otro parámetro que hay que ajustar. Un título corto es poco adecuado, pero los buscadores pueden pensar que les

estas haciendo spam si encuentran uno demasiado largo. Hay muchas consideraciones al respecto, pero podemos indicar que lo más adecuado es que sea de **15 a 20 palabras de longitud, o del orden de los 50 a 100 caracteres**.

Un título apropiado para el mecánico de Vallecas sería:

Taller mecanico de coches en Vallecas, Madrid, Todo Auto. Reparacion de lunetas limpiaparabrisas electricidad recambios aceite

Primeros párrafos

Es otra cosa en la que se fijan mucho los motores de búsqueda al indexar una página web. Si nuestra web lo permite, debemos colocar un primer párrafo que sirva de **introducción a todo lo que podemos encontrarnos en la página** o en la web. Este primer párrafo,

- Deberá estar colocado dentro del <BODY>, lo **mas cerca posible** de este. Si es posible, evitaremos que entre la etiqueta y el primer párrafo existan otras etiquetas, como imágenes.
- Deberá **ser legible**. Este texto se verá en la página, igual que cualquier párrafo, por eso debe leerse bien.

Podríamos poner en la página del mecánico este texto después de la etiqueta <BODY>

<H1 align=center> Todo Auto</H1>

En Vallecas, situado en la carretera de Valencia a 3 km de Madrid, se encuentra nuestro taller de reparación de automóviles y recambios. Estaremos gustosos de atenderles y ofrecerles nuestros servicios de todo tipo para su coche, como cambio de aceite, reparación de lunetas termicas, y otros.

De manera adicional, y sobretodo durante los primeros párrafos podemos utilizar otras técnicas, de más dudosa utilidad.

- **Utilizar los encabezamientos** (etiquetas <H1> y similares) para destacar los títulos de las páginas siempre que se pueda.
- Definir los atributos **ALT de las imágenes** con palabras claves

Estas técnicas han de usarse siempre con moderación, evitando cualquier uso excesivo que acabe perjudicando la posición del sitio.

Etiquetas META

Muchos buscadores, sobretodo los motores de búsqueda, leen determinadas etiquetas en la página para encontrar la descripción y las palabras clave de esta, se tratan de las etiquetas META. Son de vital importancia para estar bien catalogados en los buscadores, se describirán con detalle en el siguiente capítulo.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Etiquetas META

Muchos buscadores, sobretodo los motores de búsqueda, **leen determinadas etiquetas en la página para encontrar la descripción y las palabras clave** que se le asocian. Estas son las **Etiquetas META**, o META Tags.

Es muy recomendable construirlas de manera adecuada, aunque no todos los buscadores hagan uso de ellas. En concreto, los índices no suelen consultar estas etiquetas, ya que extraen esta información generalmente de los formularios que rellenamos para registrar la página.

Las etiquetas META se sitúan **en la cabecera del documento HTML**, entre las etiquetas <HEAD> y </HEAD>. Como se indico anteriormente, se han de introducir con estas etiquetas tanto la descripción como las palabras clave, cada una con una etiqueta distinta.

La etiqueta META con la descripción tiene esta sintaxis:

```
<META name="description" content="descripcion de la página aquí">
```

Mientras que **la etiqueta META con las palabras clave** tiene esta otra forma, muy parecida:

```
<META name="keywords" content="palabra clave 1,palabra 2,palabra3">
```

Ahora veamos algunas recomendaciones para construir estas etiquetas.

La descripción

- Puede ser parecida al título, si cabe un poco más **descriptiva y siempre debe ser una frase lógica** completamente legible.
- El tamaño puede estar **entre 150 o 200 caracteres**.
- **No debemos repetir la misma palabra** en la descripción, pues pueden pensar que hacemos spam.

Una descripción adecuada para el mecánico de Vallecas podría ser la siguiente: **Taller mecánico de Vallecas, Madrid. Reparación de vehículos, lunetas térmicas, cambio de aceite y otros servicios. Distribuidor autorizado Seat - Audi - Volkswagen.**

Las palabras clave

En la etiqueta META donde deberían ir las palabras clave, nosotros recomendamos colocar **frases clave**. Una detrás de otra, por orden de importancia.

- **No se debe repetir una frase clave.**
- Una misma **palabra no se debe repetir más de 5 veces**, en distintas frases claves, se entiende. Esto es debido a que si la repetimos excesivamente el motor de búsqueda puede pensar que estamos haciendo spam. Las palabras como "el", "de", "y", y otras parecidas generalmente no serán tenidas en cuenta por los motores de búsqueda, y no pasara nada por repetir las más de 5 veces.
- Como se ha visto, las frases clave han de estar **separadas por comas**. Muchos buscadores tratan a las comas de la misma manera que los espacios en blanco, con lo que si ponemos dos frases clave seguidas que tienen sentido si las leemos como si fuese una, tanto mejor. Conviene, por esto

último, que no tengamos dos frases clave con casi las mismas palabras muy próximas entre sí.

- El tamaño recomendado para el texto con las palabras clave está **entre 200 y 400 caracteres**. No es problema de todos modos en excederse, pues cada buscador leerá un determinado número de palabras y las del final las desechará. Lo que sí es problemático es repetir muchas veces la misma palabra clave, como ya se ha dicho.

Para nuestro ejemplo, podríamos poner todo este texto como palabras clave: **taller coche Vallecas Madrid, reparación vehículo Vallecas Madrid, mecánico de automóviles Vallecas Madrid, recambios para vehículos Madrid Vallecas, servicios reparación de automóviles Madrid Vallecas, taller reparación de coches, lunetas térmicas para automoviles, cambio de aceite, mecánico y recambios para vehículos, mecánico coche servicios automóvil**

Otras etiquetas META

La descripción y palabras clave no son las únicas etiquetas META que podemos incluir en una página web, aunque sí las más importantes. Otros ejemplos de etiquetas META son el autor de la página, el idioma, email de contacto, etc. Podemos ver estas etiquetas y cómo se construyen con nuestro [generador de etiquetas META](#), que puede ser muy útil para crear las de tu página web sin ningún esfuerzo. En el siguiente capítulo vamos a reunir todas estas ideas y ver cómo se construiría una página web que las contemplase.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Todo el HTML

Veamos ahora un ejemplo de **página HTML que recoge todo** lo dicho anteriormente sobre el título etiquetas META y primeros párrafos.

```
<html>
<head>
<title>Taller mecanico de coches en Vallecas, Madrid, Todo Auto. Reparación de lunetas
limpiaparabrisas electricidad recambios aceite</title>
<META name="description" content="Taller mecánico de Vallecas, Madrid. Reparación de vehículos,
lunetas térmicas, cambio de aceite y otros servicios. Distribuidor autorizado Seat - Audi -
Volkswagen.">
<META name="keywords" content="taller coche Vallecas Madrid, reparación vehículo Vallecas
Madrid, mecánico de automóviles Vallecas Madrid, recambios para vehículos Madrid Vallecas, servicios
reparación de automóviles Madrid Vallecas, taller reparación de coches, lunetas térmicas para
automoviles, cambio de aceite, mecánico y recambios para vehículos, mecánico coche servicios
automóvil">
</head>
<body>
<H1 align=center> Todo Auto</H1>
En Vallecas, situado en la carretera de Valencia a 3 km de Madrid, se encuentra nuestro taller de
reparación de automóviles y recambios.
Estaremos gustosos de atenderles y ofrecerles nuestros servicios de todo tipo para su coche, como
cambio de aceite, reparación de lunetas térmicas, y otros.
<br>
<br>
Resto de la página web...
</body>
</html>
```

Así es como debería quedar la página, requiere bastante trabajo, pero los resultados serán suficientes para compensarlo.

En el siguiente capítulo vamos a conocer otros trucos para el registro, que nunca debemos utilizar, pues resultan cotraproducentes para nuestros objetivos.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Falsos trucos

Además existen una serie de trucos adicionales, que se pueden utilizar y lo más seguro es que sean contraproducentes. Los destacamos aquí para que **NO los pongáis en práctica** pues pueden trataros de pagina spam, con el consiguiente perjuicio en su posición entre los resultados.

- Poner un **párrafo entre comentarios HTML** con la descripción del sitio, o repitiendo palabras clave de la página.
- Poner un **texto del mismo color que el fondo** (con lo que no se ve) con las palabras clave.
- **Repetir excesivamente** las palabras clave en el texto.
- Utilizar **palabras clave que luego no figuren** por ningún sitio **en la página**.
- En general, **cualquier técnica que sea fraudulenta** para conseguir estar mejor situados, pues poco a poco se van descubriendo estas técnicas y los buscadores las van penalizando a medida que las detectan.

Como se debe haber comprendido, al final, lo más recomendable es realizar la página, la creación de etiquetas META y demás, de una manera legal, **procurando facilitarle las cosas al buscador y sin abusar en el uso de técnicas** para situarlo más arriba en los resultados.

Esperamos, con los primeros capítulos de este manual, podáis registrar páginas y que estas se encuentren bien posicionadas en los buscadores. En próximos capítulos vamos a tratar de describir otras técnicas de promoción distintas a los buscadores.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Otros métodos de promoción

Podemos hacer muchas cosas para que nuestra página sea más popular, básicamente se trata de hacer que la dirección de nuestra página figure en mayor número de sitios. Vamos a proporcionar a continuación una serie de ideas:

Primero es aconsejable que **conozcáis el medio en el que os encontráis**. Existen en Internet muchas ideas para promocionar una página, muchos lo están haciendo muy bien, incluso entre las páginas que tratan vuestro mismo tema

podéis encontrar ideas para enriquecer vuestra web. Por otro lado también podéis aprender así de los errores de otras personas, y no cometerlos vosotros, por supuesto.

Poner la **dirección en la firma de vuestros correos electrónicos**. Así tendréis un posible acceso con cada correo que mandéis. Si incluís una breve descripción del sitio tanto mejor, pues puede aumentar el interés por la página.

Del mismo modo, es aconsejable que **pongáis en otros lugares la dirección** de la página. Si se trata de una empresa, por ejemplo, se puede poner el URL en el membrete del papel de oficina, en los sobres, carpetas, etc.

Daros un paseo por las **listas de correo o las news de temas relacionados** con vuestra página para sugerir la dirección. Esto se debe hacer con cautela, y solo cuando puede servir nuestra página para algo útil: No debemos presentar nuestra dirección como si fuera publicidad puesto que, por un lado, no conseguiremos que la gente se fije en nosotros y pulse en nuestro enlace, y por otro, conseguiremos el efecto contrario al deseado al crear animadversión contra la página por estar haciendo spam a los usuarios de la lista o news.

Es muy importante que nuestra página **figure en índices especializados en la temática de nuestra web**. No son páginas tan importantes como los buscadores, pero reciben visitas que pueden estar muy interesadas en nuestro sitio. Por ejemplo si es un manual, este debe figurar en todos los índices de manuales que se pueda, como www.ciberteca.net y otros más modestos que existen. Si es un recurso para crear páginas web, debería figurar en el buscador de www.desarrolloweb.com y en otras listas de recursos que existen.

Siempre es buena idea **intercambiar links con páginas que traten la misma temática**, para crear tráfico de unas a otras. No hay que mirar a las demás páginas como competidoras sino como enriquecimiento de vuestros contenidos o servicios

También debemos preocuparnos de que las **páginas personales de los usuarios, que muchas veces tienen listas de links**, contengan el nuestro. Sería adecuado crear un pequeño **botón para sugerirles a estos que lo incluyan en su sitio** e indicarles cómo deben hacerlo.

Anunciarla en otros medios de comunicación. Estamos sugiriendo que se haga publicidad en televisión, radio, prensa, etc. Esto, lógicamente, no está al alcance de todos pero evidentemente sería útil, sobretodo porque esta publicidad se encuentra "fuera de la Red" y puede llegar a otro tipo de personas distintas de las que podemos llegar a través de ella.

También podemos **promocionarla fuera de Internet** con otros medios más económicos: en el pueblo donde vivo he recibido **panfletos** donde se anuncia una web de servicios relacionados con esta población. Estos no son muy caros y además en target es de lo más apropiado. También se me ocurre colocar carteles en lugares frecuentados por esta gente con la dirección de la página y las "delicias" que ofrece, como las paradas de autobús.

Se puede, y esto es gratis, **enviar notas de prensa a los distintos medios** comentando la creación del sitio. Si alguno de ellos se hace eco de esta noticia ganaríamos muchas visitas. Del mismo modo, podemos mandar nuestra dirección a **revistas de Internet**, que suelen tener una sección donde presentan webs atractivas, y en muchos casos, las páginas personales de sus lectores.

Montar una **lista de correo** para que la gente que visite la página se pueda apuntar. Así podemos conseguir una buena cantidad de personas interesadas en nuestro web y **mandarle novedades del sitio**. Con cada correo que enviamos tenemos muchas posibilidades de conseguir una visita.

También podemos esforzarnos un poco más para **crear un boletín que podemos mandar a los componentes de una lista periódicamente**. Este boletín puede contener noticias del día, reportajes, direcciones que ver...

El **intercambio de banners** es otra buena manera de incluir referencias a nuestra página. Para crear un banner debemos ser muy cuidadosos y hacerlo de manera que este sea lo más llamativo posible. Se pueden encontrar muchas páginas de intercambio en [nuestro buscador](#). También podemos **pagar por exposiciones de nuestros banners** en otras páginas, o por clics. Pero esto solo vale para el que se lo pueda permitir.

Otra posibilidad para conseguir visitas es estar **catalogado dentro de hits**, o lugares donde se presentan las páginas más visitadas. Normalmente para acceder a estos hits debemos registrarnos y únicamente competimos entre las páginas que también se encuentran registradas.

Para ir acabando, también se nos ocurre crear un **sistema de afiliados** de cualquier tipo. Esto es que nuestra página ofrezca dinero otros por promocinarnos de alguna forma, por ejemplo, teniendo nuestra página como página de inicio en los exploradores, pagando a los usuarios por que pongan nuestro servicio de búsqueda en su página, o por usuario registrado a nuestro servicio que viniese de su web.

Como se puede comprobar, existen **multitud de vías para la promoción** de un sitio. Nosotros hemos enumerado unas cuantas, reconocemos que algunas son un poco rebuscadas o no están al alcance de la mayoría, pero hay que contar con **otras tantas que se podrían añadir con un poco más de imaginación**.

Por último, señalar que otra labor muy importante **es seguir promocionando el web, sin parar** de hacerlo nunca. Nuevos esfuerzos en promociones del web se traducen en usuarios que te recuerdan y grupos de gente que aun no te conocían por no haber llegado a ellos el mensaje anteriormente.

En el siguiente capítulo veremos algunos consejos para que nuestro web esté preparado para asimilar todos nuestros esfuerzos en su promoción.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Consejos para el web

Es importante que nuestro web pueda asimilar todos los esfuerzos que hacemos en promocionarlo, para que estos se traduzcan en usuarios satisfechos y nuevas visitas cuando vuelvan. Al final, lo que te da tráfico de verdad en la web es que la gente vuelva a menudo a ella porque le estás ofreciendo algo que le resulta útil, y se lo estás ofreciendo en bandeja de plata.

Veamos algunos consejos que debéis seguir y nuevas ideas para vuestros webs.

Renovar la información cada cierto tiempo, incluyendo nuevas páginas con nuevos contenidos. Además es importante hacerle llegar **a la gente el mensaje de que estás dedicándote a ello**, poniendo, por ejemplo, una sección donde se muestran las novedades del sitio o gráficos indicando los recursos que son novedad en el web.

Revisar tu web en algún servicio de los que te dan informes sobre su estado, por ejemplo www.websitegarage.com. Estos sitios te pueden chequear el web e informar de las cosas que estás haciendo bien o mal en variados aspectos como la compatibilidad con los navegadores, si está bien hecha para una correcta indexación en los buscadores, si tienes algún link roto, etc.

Poner en algún sitio visible una **página de contacto con el visitante**, donde puedan realizar preguntas, sugerencias u otro tipo de comentarios. No se os olvide contestar a todos los correos que recibais. En desarrolloweb hemos publicado un [reportaje sobre contacto con el navegante](#) que te puede ayudar en este sentido.

Ofrecer servicios desde tu web es otra forma de hacer que la gente vuelva a menudo. Si puedes poner cualquier cosa como un buscador, correo gratuito, un servicio de alertas, estarás haciendo que muchos usuarios vuelvan a la página para utilizarlo.

En general, como hemos ido diciendo, se trata de tener una página web sin errores clara, con buen diseño y con contenidos y servicios interesantes, es decir, **una página web cuidada al detalle**.

En el siguiente capítulo vamos a hacer una recapitulación de todo lo visto hasta ahora en este manual y a proponeros varias herramientas para la promoción.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Recapitulación

Haciendo una recapitulación de lo visto hasta ahora en este manual, vamos a proponer unas herramientas para la correcta y rápida promoción del sitio y a ofrecer en un único archivo todos los textos de los distintos capítulos que llevamos.

Servicios

En desarrolloweb.com disponemos de dos servicios útiles para promocionar tu página.

MULTIREGISTRO EN BUSCADORES

www.desarrolloweb.com/promocion/multiregistro

Es una herramienta que te permite registrar una web en varios buscadores a la vez con solo introducir una vez los datos de la página. Con una clasificación de los buscadores por distintos criterios, para que te registres en aquellos que te interesan. Revisada constantemente para que todo funcione bien.

CREACIÓN DE ETIQUETAS META www.desarrolloweb.com/etiquetasmeta

Generador de etiquetas META para tu web. Para que tengas las etiquetas META en tu página web sin errores y no sea una excusa el no saber hacerlas para no

tenerlas.

Texto del manual

Esperamos que para vosotros, promocionar una web de manera correcta en los buscadores, y en general, por todos los medios posibles, no sea un misterio. De todos modos os ofrecemos ahora un texto que recoge todo lo escrito en el manual hasta el momento, para que lo podáis imprimir, tenerlo a mano y leerlo con calma.

MANUALDEPROMOCION.ZIP 12 Kb

Este texto lo podéis tomar de manera gratuita, por supuesto. Pero os recordamos que pertenece a esta web y que **cualquier uso distinto al particular no está permitido**, ya que viola los derechos de autor.

Este manual se toma un descanso aquí, pero sin duda nuevos capítulos se le irán añadiendo a modo de apéndices que profundicen en temas más específicos.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Cómo ayuda un dominio

Disponer de un **dominio propio** para la web que queremos promocionar puede ser de gran utilidad, ya que le dará a nuestras páginas un nombre que las identifique perfectamente en la Red, de una manera personalizada y muy profesional, **con un aire de "marca en Internet"**.

Escoger bien el nombre del dominio es fundamental, si este tiene gancho podremos ser **fácilmente recordados**, y conseguir muchas ventajas:

- Nuestros visitantes pueden volver **fácilmente a nuestro sitio** con solo introducir una dirección URL corta y sencilla.
- **Promueve la imagen** del producto, servicio o empresa en Internet y le da un valor adicional.
- Nuestra web tiene un nombre equiparable al de otras grandes empresas y proyectos en Internet. **Estamos al mismo nivel que los mejores.**
- Cuando se ve un URL que únicamente consta del dominio en cuestión en los buscadores se piensa, en un principio, que se ha encontrado una página con **mucha información** del tema que se buscaba.

Por el contrario a lo que se podía pensar, tener un dominio con las palabras claves de nuestra temática no nos va a ayudar a estar mejor situados entre los resultados de la búsqueda - para estar más arriba entre los resultados hay que seguir nuestros consejos explicados al principio de [este manual](#) -. Es decir, si nuestra dirección web es www.manuales_de_diseno_web_html_css_asp_php_java.com no tiene por que estar posicionada mejor que si se llamase www.geocities.com/petardo, ya que los **buscadores usualmente no miran en la dirección de la página para extraer las palabras clave** ni ningún otro contenido que la haga más representativa con respecto a un tema.

Para obtener más información sobre los dominios en Internet, los distintos tipos, métodos de registrarlos, y donde conseguir dominios gratuitos se puede consultar

el reportaje [Los dominios, cómo registrarlos](#), publicado en desarrolloweb.

También resulta interesante, llegado a este punto, un artículo que se ha publicado dentro de este mismo manual relativo a la [elección adecuada de un nombre de dominio](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Profesionales I: Promocionar proyectos

Como dar a conocer un proyecto a los medios off-line.

Debemos distinguir 3 tipos de canales de comunicación:

- Publicidad
- Información inducida (comunicados de prensa)
- Noticias

La seducción se afianza en el primer nivel

La manipulación en el segundo.

La **credibilidad** se afianza, sólo, en el tercero.

En los 60 Mac Luhan (Marshall) lanzo la frase "El medio es el mensaje", me podéis llamar nostálgico, pero creo que es totalmente vigente.

Según el canal que se utilice no solo aumenta o disminuye el valor de la información, sino que cambia la naturaleza misma del mensaje transmitido.

Una de las especificidades de la información, a diferencia de otro tipo de producto de la economía tradicional es la siguiente:

Todo producto tiene un valor de uso y un valor de cambio; ejemplo: el agua a diferencia del oro tiene una gran valor de uso y un escaso valor de cambio.

La información sólo adquiere valor cuando se usa. Por tanto la adjudicación de valor dependerá de quien la use (o sea del receptor).

Una misma información puede tener gran valor para una persona y ninguno para otra.

Es decir, no hay forma objetiva de asignar valor a la información, ya que el valor lo da el sujeto receptor de acuerdo con sus necesidades y circunstancias.

Como no es posible probar la información sin acceder al contenido, la única referencia de valor para el receptor es la identificación que se produce entre valor del contenedor de la misma y el valor del contenido.

Dicho de otra forma, el valor de la información viene determinado por el valor de su contenedor.

Cuando el receptor al que no dirigimos es un periodista debemos tener en cuenta sus especificidades profesionales.

- El criterio informativo siempre será suyo, es un receptor activo
- Los periodistas aceptan el valor de determinadas fuentes y desprecian el de otras. (*en el argot periodístico información de "buena fuente"*).
- Están literalmente bombardeados por centenares de mensajes interesados.
- Su criterio de adjudicación de valor dependerá de la naturaleza misma del mensaje, su riqueza, su alcance, y muy especialmente de la fuente que lo transmite.

Para establecer una buena política informativa debe existir una perfecta correlación entre:

- Mensaje
- Canal
- Fuente

Cuando alguno de estos factores falla o no encaja con el resto, el sistema se cae.

Consejos para una start-up que quiera establecer una política de comunicación en medios tradicionales:

- Comience conociendo los medios de proximidad a su negocio, el periódico, la emisora local, establezca con ellos una relación de confianza.
- Localice a los corresponsales locales de los grandes medios, trabaje en la misma línea.
- Utilice la prensa del sector, siempre más sensible a esta tipología de proyectos.
- Si la situación de su negocio se lo permite contrate los servicios de un gabinete de prensa solvente.
- Desconfíe de los servicios de emisión de comunicados de prensa indiscriminados, la relación personal con el periodista es mucho más eficaz.
- Huya de los proyectos de comunicación de no se estructuren sobre los canales tradicionales de recepción de noticias. (Ejemplo, los Webs de comunicados de prensa son poco eficaces).
- Los periodistas, en nuestro país, tienen fundamentalmente una pantalla de referencia que es la que esta conectada con Agencia EFE. Cualquier medio alternativo o paralelo carece de valor.

Fiebre comunicadora en la llamada nueva economía.

El binomio sagrado: "Hacerlo bien y hacerlo saber"

Muchas veces, especialmente en el momento actual, influidos por esta fiebre comunicadora no dejamos llevar por el segundo factor del binomio.

Participando de la inflación comunicativa conseguimos que "comunicar" se vuelva un "verbo intransitivo". (Ignacio Ramonet)

A este mal uso contribuye hoy el círculo perverso que se establece entre burbuja financiera y burbuja informativa.

Estamos convencidos que cuanto mayor sea nuestra presencia mediática mayor será nuestro valor.

Entramos en una cultura en la que el simulacro se vuelve potente (incluso, no lo niego, a veces productivo)

A esta situación Freud la calificaría de delirante.

Y nos instalamos todos en este delirio colectivo, mediante el cual una imagen bien orquestada adquiere más valor (bursátil) que un proyecto sólido.

Una start-up el énfasis debe ponerlo en "hacerlo bien", basándonos en tres factores:

- Producto sólido
- Trabajo
- Músculo (conceptual y financiero)

A esto debe añadirse una gestión del proyecto eficaz y sostenida en el tiempo.

Y convertir esto en un círculo retroalimentado permanentemente.

El énfasis en comunicación en una start-up debe dirigirse principalmente al acercamiento selectivo y gradual a los inversores. (angels investors, seep capital, estructuradores financieros, private equity, venture capitalist, etc.)

En este sentido son especialmente valiosas las iniciativas como forum.org y gentemprendedora, que no sólo propician este tipo de contactos, sino que asesoran tanto en el diseño proyecto como en su estrategia de presentación a posibles inversores.

La cultura del "pelotazo" afortunadamente esta desapareciendo y emerge un nuevo sentido del valor trabajo, de los proyectos sólidos y de la gestión eficaz..

Deben denunciarse las campañas perfectamente orquestadas, en connivencia con círculos mediáticos, en las que la información tendenciosa, la imagen de diseño y el marketing, crean un sobrevalor al generar en el mercado falsas expectativas sobre determinados proyectos.

El nuevo modelo de desarrollo en el que nos encontramos llamado, interesadamente, "nueva economía" debe afianzarse en valores menos volátiles y recuperar de forma renovada el viejo concepto de la producción.

Profesionales II: Promoción en la Red

¿Publicidad on o off line?:

La publicidad off line es cara y escasamente efectiva, en un país donde el 23 % de la población no conoce ni ha oído hablar de Internet y el 60 % ha oído hablar pero no lo utiliza nunca.

Debe aprovecharse la interactividad y las posibilidades de la propia red para promocionar el negocio.

Es más rentable invertir en publicidad on -line, especialmente cuando se disponen de escasos recursos las acciones on -line permiten rentabilizar mejor la imaginación (recurso escaso pero gratuito, cuando se posee)

Tipos de promoción en la red :

Primer consejo: darse de alta en el buscador (servicio gratuito).

Los buscadores araña realizan automáticamente el alta del site, registrando automáticamente las páginas de internet.

Problemas: el registro de nuestra web puede aparecer el número 586 de la lista de resultados y las posibilidades de ser visitado son mínimas.

Soluciones al problema:

Realizar un alta escogiendo cuidadosamente las palabras clave, la sección donde aparecerá, la descripción del web, colocar bien los meta tags y aprovechar las particularidades de cada buscador.

Actualizar las altas de forma periódica.

Pagar al buscador para mejorar nuestra posición en la lista de recursos.

MARKETING VIRAL:

Como conseguir que mi site se convierta en un virus que se reproduzca automáticamente en la red, via e-mail por ejemplo (campañas piramidales de producto Ericsson, Nokia, ofrecer correo electrónico gratuito, avisos a móvil, etc)

Que debemos tener en cuenta al diseñar una campaña en la red:

Planificación de Medios en Internet

Proceso:

Definición de objetivos y presupuesto de la campaña.

Selección de soportes.

Planificación y compra.

Seguimiento de campaña.

Reconstrucción de campaña.

Objetivos y estrategias publicitarias

Generación de tráfico

Branding

Generar respuesta directa

Venta (e-commerce)

Generación de base de datos

Creación de comunidades

Generación de tráfico

Aportación a través de campañas de banners de público al site del anunciante desde un site de alto tráfico donde se encuentre representado el público objetivo.

El banner/promoción es la primera etapa de la pieza publicitaria, ha de generar el interés de la audiencia para conseguir la acción.

Hay que considerar el conjunto banner/site como una única pieza de comunicación.

El ratio medio de respuesta (click through) a las campañas de banners en nuestro país rondaba el 2%, ahora es prácticamente del 0%.

Branding

La publicidad on-line es una poderosa herramienta de branding, es evidente que la audiencia que no interactúa con la publicidad on-line está recibiendo impactos publicitarios tan útiles como los producidos por los demás medios tradicionales.

Un estudio sobre la efectividad de la publicidad "on-line" realizado por iab en 1997 indica que las marcas que realizan publicidad en la red incrementan en un 30% el grado de conocimiento de la marca entre los usuarios de la misma y que la percepción sobre la marca puede mejorar un 55%.

Respuesta Directa

A través de campañas de publicidad podemos conseguir respuesta directa de la audiencia en tiempo real.

Es una gran solución para conocer la opinión sobre un producto o un tema concreto por parte del público objetivo.

e-commerce

Realizar el proceso de la venta de forma directa (publicidad/presentación de producto/venta) y en un solo paso.

Podemos realizar campañas de publicidad dirigidas a aportar compradores a una solución de comercio electrónico o que todo el proceso se realice en el banner

Objetivos y estrategias publicitarias

Promociones Puntuales

Los site de tercera generación son los sites de producto (los sites institucionales son cada día menos utilizados), microsites que tienen una vida muy limitada y que presentan un producto o servicio.

El extremo máximo de este tipo de sites son las campañas banner/site creados para difundir una promoción puntual.

El bajo coste de producción y la velocidad de la respuesta de dichas campañas hacen de internet un medio insuperable para este tipo de promociones.

Generación de Base de Datos

El coste por registro de una base de datos generada en internet es realmente competitivo (alrededor de 400.- ptas).

Es muy importante la creación de dichas bases de datos para crear sistemas de fidelización de usuarios a nuestro site o simplemente para realizar ofertas puntuales a través del correo electrónico.

Informe de **Albert Pérez Novell**
Gerente de la Agencia EFE en Cataluña

Profesionales III: Tipos de publicidad

Tipos de Publicidad on-line:

El Banner

Efectividad en claro descenso.

El formato estándar de la publicidad en internet.

A esta pieza publicitaria se le pide en la mayoría de los casos que genere tráfico hacia el website del anunciante.

Hay que valorar su potencialidad como herramienta de branding.

El formato más utilizado es el gif animado y las medidas 468x60 píxeles

Banners

Elementos que influyen en el éxito de la campaña:

La creatividad.

La planificación de la campaña.

El conjunto campaña de banners/website

Según el estudio "banner ad location effectiveness" (1997 athenia associates) hay dos localizaciones preferentes:

- En el tercio superior de la página
- En el lateral derecho

El mismo estudio indica que más de un banner en la misma página no mejora el ratio de click. se divide entre dos creatividades el número de clicks.

Creatividad del Banner

Mensajes cortos claros y legibles.

Crear un sentido de urgencia.

Mensajes enigmáticos.

Formular una pregunta.

Utilizar mensajes como "gratis", "gana"...

Invitar al clic.

Integrar en el mensaje del banner la invitación a la acción.

Animación de los banners.

Múltiples creatividades.

Planificación de la Campaña

Siempre que sea posible, personalizar el mensaje para cada uno de los targets a los que nos dirigimos y teniendo en consideración las posibilidades de segmentación de los soportes

Nuevos formatos.

Para contrarrestar la caída del porcentaje de respuesta de los banners en formato tradicional (gif) aparecen en el mercado nuevos formatos que aumentan la interactividad y notoriedad de los banners.

Estas tecnologías son:

banners html y dhtml
banners flash
banners en java

Aparecen nuevas propuestas publicitarias en la red que intentan potenciar la notoriedad como su efectividad de respuesta.

Estas propuestas en algunos casos topan con el problema de la falta de ancho de banda.

En otros casos, pueden ser considerados por los internautas como demasiado intrusivos.

Estos formatos son las pop-up windows, los intersticiales, los banners desplegados o los cursores personalizados

Ventanas flotantes

Aumentan la notoriedad presentándose en un navegador nuevo.

Problemas en navegación lenta.

Intersticiales

El **interstitial** intenta recrear el spot televisivo en internet.

Se enfrentan al problema de la lentitud en internet.

Pueden ser considerados por el usuario como demasiado intrusivos.

Banners Extensibles.

Una de las últimas propuestas aparecidas en la red.

El banner dispone de la posibilidad de extenderse y disponer de más espacio para plantear la oferta o diversificarla antes de remitirte a la página del anunciante.

Cursores Animados.

Introducir el mensaje o animaciones del anunciante en los cursores del ordenador del usuario.

Este sistema puede reforzar la notoriedad de banners, patrocinios....

[Comet systems](#) es la compañía que los ha lanzado al mercado.

Acciones especiales:

Ventajas del Patrocinio

Potencia la imagen de marca del anunciante.

Ofrece información de valor añadido al usuario e integrarla en un entorno que conoce.

Conoce con más exactitud el perfil de la audiencia que nos visita.

Crea bases de datos de prospects.

Audiencia asegurada desde el primer día.

Posibilita el acceso a anunciantes que no disponen de web site.

Objetivos:

Generar tráfico al site

Construir imagen de marca

Potenciar la interacción con los usuarios.

Potenciación de los contenidos de la misma.

Con una campaña de banners en los principales websites españoles, atrayendo tráfico no a las páginas del anunciante, sino a la sección patrocinada, que se hallaba en otro website.

Por contenidos del web site.

Las temáticas del site nos permiten intuir el tipo de público que visita la página.

Es UN sistema mUY utilizado actualmente.

Los costes de inserción aumentan al comprar sites con públicos muy definidos.

Existen sites nicho donde podemos encontrar targets muy definidos. un ejemplo claro sería un site de contenido para aparejadores y arquitectos.

El idioma del site nos puede segmentar públicos en ocasiones.

Por usuario registrado / datamining.

Los sites donde los usuarios están registrados permiten dirigir campañas a perfiles sociodemográficos muy precisos.

Ejemplos de estos tipos de site son msn y financial times

El seguimiento de los perfiles de los usuarios que se están realizando por parte de las grandes redes publicitarias permitirán próximamente comprar perfiles de usuarios independientemente de los contenidos del site que visite el usuario. Por el momento, estos sistemas de datamining no permiten procesamientos suficientemente rápidos para permitir servir el banner en décimas de segundo.

Adservering.

Los servidores de publicidad identifican al usuario que accede a un site y le sirven los banners dirigidos a su perfil.

De esta forma se sirve la publicidad adecuada a cada usuario.

Los adservers permiten la compra por impresiones, utilizar más de un banner en la campaña y conocer los datos en tiempo real de la evolución de la campaña.

Algunos adservers están auditados por terceras partes (abc electronics, net ratings).

Adserving.

Las segmentaciones que permiten son:

- País (dominios o sistemas mixtos ip/dominio)
- Día y hora
- Dominio
- Sistema operativo
- Navegador
- Por banner
- Frecuencias...

Cookies.

Fichero de texto instalado por el servidor web en el navegador que contiene información sobre las preferencias del usuario y otros datos diversos, que activan ciertas respuestas en los sistemas a los que se conecta. Es el sistema que utilizan los servidores de publicidad para identificar máquinas únicas, activar el sistema de frecuencias o recopilar información sobre el perfil del usuario de la máquina. (www.cookiecentral.com)

Informe de **Albert Pérez Novell**
Gerente de la Agencia EFE en Cataluña

Profesionales IV: Mercado y soportes.

mercado y selección de soportes

Grandes sites con departamento comercial propio (yahoo, olé!, ciudades netropolis...)

Versiones on-line de medios de comunicación tradicionales (el país, recoletos, idg, grupo correo...)

Redes publicitarias, exclusivistas que aportan a sites medianos/grandes tecnología de adserving y equipos comerciales para la comercialización de los espacios publicitarios (doubleclick...).

Formas de Compra:

Espacio tiempo:

Contratar un banner durante un mes en la página principal de un site.

No hay garantía de audiencia.

No hay segmentación

No se dispone de datos sobre la evolución de la campaña.

formato poco utilizado.

Por Impresiones:

Una impresión = un banner cargado en un ordenador.

Sólo se paga por la audiencia que ve nuestro anuncio.

Posibilidad de segmentación (por país...)

Resultados de campaña en 24h.

Formato de compra más utilizado.

Coste de la impresión de 3 a 10 pesetas.

Por Click:

Un click = un usuario pulsa sobre el banner y visita la página del anunciante.

Sólo se paga por la audiencia que va a nuestro site.

Posibilidad de segmentación (por país...)

Resultados de campaña en 24h.

Formato de compra poco utilizado.

Coste del click de 160 a 200 pesetas.

Por transacción:

El site cobra un porcentaje por las ventas generadas por las campañas publicitarias.

Existen soluciones de adserving que permiten controlar este tipos de campaña con total transparencia.

Seguimiento, control y valoración de los resultados de campaña

Acceso a las estadísticas on-line:

Diariamente se actualizan los datos de la evolución de las campañas.

En el caso de campañas contratadas site a site es importante valorar los resultados de forma individualizada en cada uno de los soportes e intentar optimizar los resultados tras disponer de datos con suficiente peso estadístico.

Acceso a las estadísticas on-line:

Hay que valorar los ratios de respuesta según el día de la semana, la hora del día y según los diferentes banners.

La mayoría de los sistemas permiten realizar reports específicos con estas variables y realizar modificaciones de campaña en 24h.

Informe final de campaña:

Los informes de final de campaña nos permiten valorar si hemos conseguido los resultados definidos en la estrategia de campaña.

La valoración se ha de realizar de forma conjunta con la información del servidor de publicidad, el análisis de los logs y los resultados cuantificables de respuesta directa.

Campañas de éxito:

variables

Tráfico generado hacia el site (cuantitativamente y cualitativamente).

Prospects generados. calidad de los mismos.

Ventas generadas.

Comparación del tráfico tras la campaña con los datos anteriores al inicio de la misma.

Informe de **Albert Pérez Novell**
Gerente de la Agencia EFE en Cataluña

Popularidad de un sitio web

La popularidad de un sitio es un concepto muy amplio, en general mide lo conocido que sea entre los internautas. En el caso de páginas especializadas, como desarrolloweb, su popularidad significa lo conocido que sea entre los usuarios que se interesan por el desarrollo de páginas.

Cuando nosotros promocionamos un sitio procuramos que este sea conocido, buscamos su popularidad. En muchos casos, el simple hecho de que nuestro sitio sea popular nos ayudará también a estar mejor considerados entre la oferta de sitios de Internet. Los buscadores nos admitirán las páginas con mayor probabilidad y nos posicionarán mejor entre los resultados.

Por todo esto, debemos dar una imagen de "sitio popular", esto se consigue definiendo una buena imagen de marca y, por supuesto, ofreciendo buenos contenidos y servicios.

Los buscadores ponderan según la popularidad

Existe una nueva generación de buscadores que ponderan los resultados de sus búsquedas en función de lo popular que sea un sitio. Éstos realizan un sencillo cálculo que consiste en contar los enlaces que tienen otras páginas hacia un sitio web determinado. Los sitios que tengan más enlaces dirigidos hacia ellos tendrán una puntuación más alta y saldrán más arriba en los resultados.

Pongamos un ejemplo: buscando la página de la petrolífera Repsol-Ypf podríamos introducir repsol ypf en el buscador. Éste, buscará entre sus páginas y extraerá muchas webs con estas palabras. Una de ellas será la compañía que buscamos y seguro que habrá muchas páginas que no sean la web corporativa que estamos buscando, pero muy probablemente contendrán enlaces a la página de Repsol-Ypf. Así, podremos imaginar que la página que estará más enlazada de todas las páginas de los resultados será la web corporativa, por lo tanto saldrá en el primer resultado de la búsqueda.

Medir los enlaces que hay hacia tu sitio

Para medir los enlaces que hay en otras páginas hacia tu sitio debes utilizar los propios buscadores. En muchas ocasiones tienen una herramienta de búsqueda avanzada o alguna ayuda que explica como puedes hacer esta búsqueda. Por ejemplo, Altavista incluye esta opción y para utilizarla deberemos buscar por **link: seguido por la dirección que quieres contar el número de enlaces**, de esta forma: **link:http://www.tudominio.com**

También existe la posibilidad de hacer esta búsqueda en www.linkpopularity.com a través de una página que te devuelve el número de páginas que te enlazan en distintos motores de búsqueda.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Cómo tener un web-negocio y no morir en el intento

Internet y las nuevas tecnologías han supuesto para muchas personas emprendedoras y/o PYMES un nuevo modelo de negocio. Al principio todo parecía ir bien y 'viento en popa': eran los tiempos del boom de la nueva economía y la implantación de grandes e-empresas que gastaban ingentes cantidades de dinero en publicidad, marketing,... prometiendo conquistar un nuevo "el dorado". Actualmente **la crisis del sector, la bancarrota de la bolsa y los nuevos valores tecnológicos y la desilusión de los inversionistas está produciendo un receso en el desarrollo de web-negocios exitosos.**

Hoy día crear una web con la finalidad de convertirla en una empresa capaz de generar los suficientes ingresos como para vivir de ella se ha convertido en algo sumamente complicado, así que ahí van algunas sugerencias para todos aquellos emprendedores/as que tienen una idea, quieren llevarla a Internet... pero no quieren morir en el intento.

1.- Reunir una cantidad de dinero suficientemente importante como para afrontar, al menos, dos años de gastos. Esa idea de que en Internet "de la noche a la mañana" uno es rico y puede irse a las Bahamas mientras los ordenadores trabajan por él/ella es una falacia. Sin dinero previo no hay nada que hacer.

2.- **Disponer de una idea ligeramente diferente a lo ya existente pero no extremadamente distinta.** Si vas a crear algo, que sea nuevo, o que tenga algo diferente a la competencia, pero no crees algo tan nuevo que no lo entienda nadie. ¿Dónde hay más consumidores/clientes?, ¿qué es lo que necesitan/piden?, pues ahí hay que aplicar la idea.

3.- **Contar con un equipo que te complemente.** Si eres un especialista que ha desarrollado un innovador programa pero no cuentas con una persona que se encargue del área administrativa y de marketing, tienes tres meses de vida. Nadie puede hacer por sí solo todo completamente bien, y eso es justamente lo que le pedirán sus clientes: quiero lo mejor al mejor precio.

4.- **Disponga de un saquito con bastante dinero para invertir en publicidad y marketing.** En Internet cada vez hay más productos/servicios y hay que "hacerse ver". Ni el mejor producto/servicio puede triunfar si no se dedica una cantidad de dinero mensual a la publicidad. Las estrategias ya las conocemos: banner, newsletter, email, offline, patrocinio,...

5.- **La atención al cliente es fundamental.** Ante precios similares, su cliente siempre preferirá estar con alguien que sabe atenderle correctamente (aunque sea un poco más caro): en Internet esto quiere decir "ahora mismo". Por supuesto, además cuentan todas las normas de cortesía y management que podamos aprender.

6.- **Tenga paciencia** si en los primeros meses "no vende una rosquilla". Probablemente todavía no sabe nadie que usted (su web) existe. Sea perseverante y márchese un plan de trabajo con submetas. Esto le ayudará a valorar si la evolución que sigue es la correcta.

7.- **Procure ser el mejor en lo suyo pero no se especialice demasiado,** a no ser que disponga de muuucho dinero. Ofrecer un producto/servicio de calidad, ajustado en su precio y con algo distintivo es vital pero además procure rodear su venta con otros "añadidos". ¡Ojo! Estos añadidos deben ser de interés para el cliente aunque a penas le generen beneficio a usted. Un cliente satisfecho es el mejor agente comercial que podemos tener.

8.- Si ha realizado todo lo anterior ya habrá pasado más de un año por lo que estará en el momento crítico: invierto un poco más o "tiro la toalla". No lo dude: **siempre mire hacia delante.** Es la etapa crítica para establecer con solidez el futuro de su web-negocio. Porque en Internet al igual que en el mundo tradicional, crear un web-negocio es fácil, lo difícil es mantenerse, continuarlo en el tiempo.

Informe de **Jesús A. Lacoste**
Director Soydigital.com; Todo Internet a su alcance
Mail: lacoste@soydigital.com
URL: <http://www.soydigital.com>

El banner perfecto

Resumen.

1. El banner funciona. Si bien como soporte de un modelo de negocio esta cuestionado por el ingresos que es capaz de generar, como medio de comunicacion es algo que se debe aprovechar.

2. En este artículo veremos puntos clave a la hora de diseñar nuestro banner para lograr una comunicación efectiva. No loop, si animación, menor peso, mas grande.

1. El banner funciona.

En un estudio de [Site Usability](#) se hace referencia a la ceguera al banner. El estudio muestra que los usuarios no son tan ciegos al banner como parece y la impresión del mensaje es efectiva.

Desde el punto de vista del anunciante, el banner puede funcionar. Cada vez internet resta mas tiempo a la television (evidentemente mas que los periodicos y revistas) y en internet se encuentran nichos de mercado que en el mundo real son dificiles de seguir.

Por otro lado, en internet tenemos la oportunidad de acceder a un rango de poblacion mas amplio que un medio con limitaciones geograficas, temporales, etc.

Incluido el 27.5.2001: Los datos de Site Usability sobre la ceguera al banner despertaron dudas por parte de algunos lectores por lo que se hizo una [encuesta](#) propia para saber hasta que punto este dato era cierto. La encuesta que hice da que un 25% de los usuarios recuerdan algun banner de su ultima sesion. Est dato da muy pocas esperanzas al banner actual con lo que debemos pensar en ir cambiando los banners. Otra idea es ir haciendo las paginas que dependen de ingresos por publicidad mas "ligeras" para que el banner pueda brillar lo mas posible y capte la atencion del usuario.

2. No siempre debe llevar a una pagina.

El dato que se toma para medir la eficacia de un banner es el click through. Esto es correcto pero no siempre es valido.

Los anuncios sirven para ofrecer informacion, posicionar o darle valor a una marca y para despertar la necesidad de adquirir un determinado producto. El click through esta una dimension mas alla de estos datos ya que ofrece acceso instantaneo a la adquisicion de este producto.

El problema es que muchos productos que se anuncian en banners no se pueden adquirir directamente. Un banner sobre el estreno de una pelicula puede ofrecer informacion sobre el dia del estreno y como opcion el visitar la pagina. En este ejemplo, la efectividad del banner no se mide solamente por el click through. La impresion correcta del mensaje es suficiente para justificar la inversion.

Este banner sirve para comunicar la fecha de estreno de la pelicula y como opcion de enlace nos invita a ver los grupos de discusion.



Recuerdo del banner:
43% algo.
57% nada.
Fuente: [Site Usability](#).

3. Debe llevar a una pagina de interes.

El enlace que contiene el banner debe llevar a una pagina diseñada especificamente para este fin. Si el banner lleva a la portada, el usuario puede sentirse confundido o no encontrar aquello que vio en el banner.

Si lo que estamos comunicando es la existencia de nuestro site, se puede preparar una pagina de bienvenida especial donde se explica el proyecto, objetivos, etc.

En el caso de este banner de Mixmail.com, pese a invitarnos a darnos de alta en su servicio, el banner te lleva a la portada general.



4. Banners avanzados son interesantes pero la gente no los usa por que no sabe si debe pinchar o usarlo.

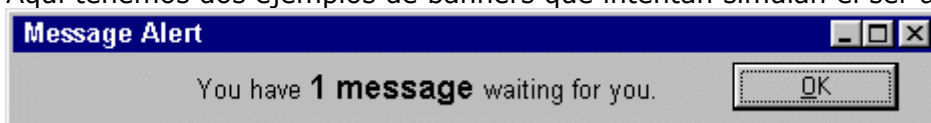
Estos banners que contienen combos, campos de texto o botones, donde el usuario puede realizar parte de la transaccion antes de ver la pagina. Estan bien en concepto, pero el problema es que mucha gente abusa de este tipo de banners, engañando al usuario haciendo parecer botones del sistema, etc.

No son muy recomendables.

Ejemplo de banner avanzado que funciona.



Aqui tenemos dos ejemplos de banners que intentan simular el ser avanzados.



5. El peso importa.

En general haciendo un bonito gif, el banner no deberia pasar de 4kbs.

Esto es evidente, pero hay que recordarlo. Haz tu banner ultra ligero. El modelo google de banners en formato texto es el paso mas avanzado en este campo. Quizas sea algo extremo, pero en el mundo google funciona (academico, profesional, usuarios avanzados). Personalmente creo que es otra dimension de la publicidad a la que mucha gente no llega.

Este es el ejemplo de banner que se puede encontrar en google.com

Marketing - Free Access to Tips, Ideas, Resources
www.MarketItRight.com Step-by-Step Guides - Launch Your Campaign! - Free - Click Here

Sponsored
Link

Imagen reducida, [este enlace permite verla a tamaño real.](#)

Si bien este modelo de publicidad es demasiado avanzado, lo que si hay que destacar de google es la [forma de comprar publicidad](#). Todo en formato web bien sencillo. Aqui puedes ver la [opinion](#) de gente que ha [revisado](#) este sistema.

6. Animacion si, pero procura fijar el mensaje desde el primer momento.

Que el gif este animado es parte de la "sorpresa" que ayuda a fijar el mensaje. Procura que la animacion ocurra en una parte del banner mientras el resto permanece estatico ofreciendo claramente el mensaje que quieres transmitir.

Pese a esto no se deben hacer loops y se debe fijar el mensaje en el primer frame.

Los loops hacen que el mensaje se distorsione al perder la identificacion de "conclusion" e "inicio". Si la gente no sabe donde esta el final, no entenderan cual es el sentido del banner con lo que parte de la efectividad se puede perder.

Desde el primer frame se debe fijar el mensaje. Muchas veces la parte superior de la pagina esta presente unos segundos hasta que el usuario hace scroll o cambia de pagina. No pierdas esos segundos contando una historia.

Aqui tenemos un par de banners donde la animacion que conduce al mensaje es muy larga.



Estos dos banners de Marca (periodico deportivo español) fallan en casi todos los puntos que hemos visto hasta ahora. Son un loop infinito, no fijan el mensaje, la duracion es exesiva, etc.

7. Procura que el mensaje no solo sea el logo.

Todavia quedan algunos banners donde el frame (entiendase como fotograma) final es el logo de la compañía anunciante. Esto es un error. El frame final debe servir para ofrecer toda la informacion necesaria para trasmitir el mensaje. El incluir la url suele ser parte de este mensaje.

El frame final puede ser imprimido, enviado por mail y debe seguir transmitiendo el mensaje con claridad.

8. El banner puede ser un soporte de modelo de negocio.

Lo que falla ahora mismo es el formato. Banner que no son rectangulares y que no estan en la parte superior de la pagina.

La busqueda de formatos y tecnicas que permitan crear una mejor impresion del mensaje en el usuario es un camino a explorar. Tambien hay que contar con la dimension que internet añade donde el mensaje se puede continuar hasta la compra directa del producto.

En un estudio de [webreference](#) se vio como colocando el banner un tercio mas abajo de lo normal, el click through se incrementaba un 77%.

Seguramente [banners gigantes](#) con flash tipo CNET sean un camino a seguir.

Permiten explorar internamente el banner, ofrecen mas informacion y no hace falta abandonar la pagina en la que estamos con lo que podemos seguir con nuestra tarea. Lo que tiene de bueno este banner es el concepto de "poner un trozo de mi web en la tuya". El banner ya no es un anuncio, si no que pasa a ser un trozo de mi web donde te puedes completar una accion sin abandonar el site en el que estas actualmente. Esto tiene mucho sentido si pensamos en la compra de productos. No hace falta abandonar el site en el que aparece el anuncio, puedo comprar el producto directamente sobre el anuncio.

Enlaces relacionados:

- ♦ Site Usability. <http://wsupsy.psy.twsu.edu/surl/usabilitynews/2S/banners.htm>. Somos ciegos a los banners.
- ♦ Webreference. <http://webreference.com/dev/banners/onethird.html>. Resumen: Based on the above results, we conclude that the placement of the ad 1/3 down the screen, increased click-through 77% (for the Photodisc ad). The Webreference ad showed the same trend, but the result was not statistically significant.
- ♦ Useit. <http://www.useit.com/alertbox/9709a.html>. Resumen: "... only the top 0.01% of websites can generate sufficient revenues from advertising: in the larger picture, advertising is almost irrelevant for the success of the Web."
- ♦ Useit. <http://www.useit.com/alertbox/9704b.html>. Resumen: "... In fact, the model predicts that the largest website will run at a rate of about 200 billion pageviews per year by the end of the Year 2000... "
- ♦ BannerTips. <http://www.bannertips.com/bannerdesign.shtml>. Banners design. Consejos de todo tipo.
- ♦ BannerTips. <http://www.bannertips.com/makersForHire.shtml>. Listado de empresa que recomiendan para crear tu banner.
- ♦ Listado de los banners mas imprimidos.
<http://209.249.142.27/nnpm/owa/NRpublicreports.topbannerweekly>

Informe de **César Martín**
 Mail: iworkwithyou@yahoo.com
 URL: <http://alzado.org>

La inteligencia emocional aplicada a Internet

Para lograr el éxito en Internet hoy día no basta con poseer un alto coeficiente intelectual (CI) y unos conocimientos técnicos muy grandes. Los cambios del siglo veintiuno están siendo testigos del predominio del coeficiente emocional (CE), un concepto nuevo que incluye el autoconocimiento y autodominio, el celo y la persistencia, la capacidad de motivarse uno mismo y de lograr resultados en conjunto con otros.

Todavía quedan algunas personas que desechan del todo la importancia de las emociones; las ven como un campo minado que se debe evitar a toda costa. En muchos casos son precisamente esas personas que, con su énfasis en los números fríos y el balance de utilidades, están más desconectados del motor del corazón que impulsa al capital humano y produce el trabajo creativo exponencial que se requiere para que una organización vaya a la cabeza en su campo en Internet.

La Inteligencia emocional es una forma de interactuar con el mundo que tiene muy en cuenta los sentimientos, y engloba habilidades tales como el control de los impulsos, la autoconciencia, la motivación, el entusiasmo, la perseverancia, la empatía, la agilidad mental, etc. Ellas configuran rasgos de carácter como la autodisciplina, la compasión o el altruismo, que resultan indispensables para una buena y creativa adaptación social.

Por otra parte no hay que olvidar que el éxito en Internet depende del más fundamental de todos los principios comerciales: satisfacer al cliente/visitante. Para ello, teniendo en cuenta las claves que nos aporta el estudio de la Inteligencia Emocional, los aspectos que debemos tener en cuenta son los siguientes:

1. **Confianza.** Generar la sensación de controlar y dominar el web site, la propia evolución y los contenidos aportados. La sensación de que los servicios/productos tienen muchas posibilidades de éxito y que usted cree en ellos debe transmitirla a sus clientes, tanto desde la propia página web como en los mensajes de correo electrónico.
2. **Curiosidad.** Tener en cuenta la sensación de que el hecho de descubrir algo es positivo y placentero. Aporte algo nuevo o una nueva visión del algo ya existente o un modo de interactuar/probar el servicio que desea vender: habrá despertado curiosidad.
3. **Intencionalidad.** Mostrar un claro deseo y la capacidad de lograr algo y de actuar en consecuencia. Esta habilidad está ligada a la sensación y a la capacidad de sentirse competente, de ser eficaz. Tenga claro cuál es su objetivo, su intención con el web y busque la manera de ser competente en su desarrollo.
4. **Autocontrol.** La capacidad de modular y controlar los productos/servicios en una forma apropiada; la sensación de control interno. ¿Es usted quien controla la evolución de sus servicios o depende de las fluctuaciones de variables externas?. Cuanto más control tenga sobre sus productos/servicios más fácil será satisfacer al cliente.
5. **Relación.** La capacidad de relacionarse con los clientes, una capacidad que se basa en el hecho de comprenderles y de ser comprendido por ellos. Póngase en el papel del cliente y piense como él ¿Qué pediría, qué necesitaría?. Fomente una relación lo más personal posible (hay que hacer humano un medio que es totalmente anónimo).
6. **Capacidad de comunicar.** El deseo y la capacidad de intercambiar verbalmente o por escrito ideas, sentimientos y conceptos con los demás. Esta capacidad exige la confianza en los clientes y el placer de relacionarse con ellos. Hable, comuníquese, no se esconda tras un perfil. Los demás le quieren oír y saber lo que piensa.
7. **Cooperación.** La capacidad de armonizar las propias necesidades con las de los clientes. Recuerde que usted busca una ganancia (económica) pero el cliente también persigue un beneficio. Y nunca olvide el soporte y ayuda post-compra.

A nivel personal, todo administrador de un web site debe plantearse la urgencia en "alfabetizarse" emocionalmente. Algunas propuestas que pueden servirle de reflexión son:

- **Autoconciencia emocional:** El conocimiento de nuestros sentimientos y sus causas nos ayuda a mejorar modelos de conducta y relaciones.
- **Automotivación:** No esperemos siempre estímulos externos. La capacidad de generarlos nos hará más independientes y capaces en situaciones adversas.

- **Intuición:** Desarrollemos nuestra propia y natural capacidad para captar e interpretar las cosas. Confiemos más en ella.
- **Toma de decisiones personales:** Nuestra opinión nos hace únicos. Atrevámonos a ser un poco más nosotros mismos.
- **Conciencia de uno mismo:** Sólo desde ella, podremos tomar auténticas decisiones y vivir de forma consciente y adulta.
- **Capacidad de manejar el estrés:** Fundamental en la vorágine de muchas vidas cotidianas.
- **Empatía:** Captar la sintonía con otras personas, aprender a ajustar el ritmo según el interlocutor, adaptarse... ayudará en cualquier contexto de relaciones humanas.

Informe de **Jesús A. Lacoste**
 Director Soydigital.com; Todo Internet a su alcance
 Mail: lacoste@soydigital.com
 URL: <http://www.soydigital.com>

Las siete reglas de Oro de un dominio

¿Qué hace de un dominio un gran dominio? ¿Cómo saber si el nombre de dominio que estoy pensando en comprar o registrar es realmente el adecuado para mí? Muy fácil, sólo hay que tener en cuenta estas siete reglas de oro.

1. Ponlo fácil. Ésta es la gran regla básica en el mundo de los dominios en particular y en el del márketing y comunicación en general. A ningún publicista se le olvida la conocida regla KISS (Keep it Simple Stupid) y desde luego nuestro nombre de dominio es la primera forma de publicitar nuestro negocio en la red.

Hay que intentar que nuestro nombre de nuestro dominio le resulte fácil de recordar al usuario. El motivo es bien sencillo. Imaginémosnos que, navegando por la red, nos encontramos con dos páginas que nos encantan: eco.com y emeronion.com. ¿Cuál es más probable que volvamos a visitar?

2. Lo breve... dos veces bueno. El dicho se cumple una vez más en el mundo de los dominios. Cuantos menos caracteres tenga un nombre, mucho más fácil de recordar y también más cómodo y rápido de teclear. Desgraciadamente, estos dominios son ya muy escasos. El 100% de los dominios con tres caracteres o menos hace tiempo que está registrado. Comprarlos puede ser una solución pero los dominios breves son, sin duda, los más valiosos por lo escaso y, por tanto, los más caros.

3. Dando sentido. Varios estudios han demostrado que las palabras con significado se recuerdan prácticamente un 80% más que aquellas que no lo tienen. Buscar un nombre que realmente diga algo puede resultar complicado pero merece la pena. Es cierto que hay empresas como Yahoo que han conseguido grandes resultados con un nombre sin significado pero ¿cuánto dinero han tenido que invertir en publicidad?

4. Sin confusión. Uno de los problemas comunes con los que se encuentran las empresas de internet es el del tráfico desviado desde su página a causa de un error de tecleo por parte del usuario. Los juegos de palabras ingeniosos pueden resultar muy divertidos y útiles para captar la atención pero, en general, son poco efectivos en la red y mucho menos si utilizamos números y guiones. El significado del

dominio "love-2u.com" puede ser muy sencillo de recordar pero el usuario se encontrará con la duda de si debía escribir "lovetoyou.com" o "loveto-u.com" o quién sabe cuántas combinaciones más.

5. ¿Somos .com? A la hora de escoger una extensión para nuestro dominio el dilema es el siguiente: ¿qué dominio se adapta mejor a nuestro mercado? Está claro que el estándar de facto en la red es el .com. Ésta es sin duda la extensión más codiciada. Pero, si no operamos a nivel internacional, también puede resultar una buena idea pensar en los dominios regionales de nuestro campo de actuación: .es, com.ar, com.mx, etc...

Otra alternativa son las extensiones .net y .org consolidadas desde hace tiempo aunque con bastante menos tirada que la .com. Si decidimos tener algo de paciencia, también podemos apostar por los .info o .biz que, aunque todavía tienen poca implantación, prometen ser importantes en un futuro.

6. ¿Qué dice de lo que hacemos? Ajustar el nombre de nuestro dominio a nuestra actividad es muy conveniente para conseguir que el usuario nos asocie con un determinado producto o servicio, y de ese modo, se pueda acordar de nosotros cada vez que lo necesite. Calor.com quizá es un gran nombre de dominio pero, si el negocio para el que lo utilizamos es una tienda de zapatos on-line, probablemente no nos va a resultar de mucha utilidad.

7. Sin malas lecturas. El último aspecto a tener en cuenta a la hora de elegir un dominio son las connotaciones asociadas al nombre que elijamos. Es importante asegurarse de que nuestro dominio no tiene connotaciones negativas que puedan quedar asociadas a nuestros productos. Y si operamos en un entorno internacional tenemos que asegurarnos de que esto no sucede con nuestra lengua ni tampoco con el resto que podamos considerar importantes para nuestro negocio. Desde luego, "estohuele.com" no sería precisamente la elección más adecuada para nuestra tienda de perfumes.

Informe de **Albert Domínguez**
Spanish Country Manager
Mail: albert@esedo.com
URL: <http://www.esedo.com>

Errores frecuentes al redactar el título de una web

Casi todos los buscadores consideran la "MetaTag Title" como la más importante del código HTML de una página web.

La MetaTag Title es además la primera información que aparece cuando se realiza una búsqueda en la mayoría de los grandes buscadores, haciéndola mucho más relevante, ya que es la primera información que el Internauta ve acerca de nuestra página web y decidirá hacer click o no dependiendo de su contenido.

Hay muchas páginas web que utilizan fabulosos títulos para crear una gran campaña de marketing en los buscadores y recibir tráfico completamente gratuito, pero también hay páginas web que ni siquiera tienen un título y se muestran como "Untitled". ¿Quién va a hacer click en un documento donde la información que aparece es simplemente "Untitled"?

Que no debes hacer al redactar un título

1- Nunca utilices una lista de palabras claves separadas por comas. Es difícil de leer

y no es profesional. Debes de redactar una frase lógica utilizando siempre las palabras clave.

2- No utilices palabras complicadas y difíciles, a no ser que tu página web sea de un tema muy específico y busques tráfico selecto.

3- No repitas en exceso tus palabras clave, una vez es suficiente.

4- No utilices siempre mayúsculas en el título, es difícil de leer. Utiliza las mayúsculas sólo cuando correspondan.

5- Dedicar tiempo a redactar un título que sea efectivo, exactamente igual a cualquier campaña de publicidad.

6- Redacta un título corto y fácil de leer. El título es como un anuncio de tu página web.

7- No olvides incluir el título en TODAS tus páginas interiores y si son títulos diferentes mucho mejor. La "MetaTag Title" debe ser siempre la primera del código HTML.

8- No utilices únicamente el nombre de tu empresa en el título. Si quieres incluir el nombre de tu empresa debe de ir siempre acompañada de tus palabras clave más importantes.

9- No utilices caracteres como !!!! o AAA para aparecer el primero de la lista en orden alfabético. Este recurso ha sido altamante utilizado y puedes ser penalizado por el buscador.

10- Por último no cometas faltas de ortografía ni palabras mal escritas que den aspecto poco profesional a tu página web. Dedicar tiempo a redactar el título de tu página web.

Informe de **Myra Palacios**
Directora de Myrosoft
Mail: myra@myrosoft.com
URL: <http://www.myrosoft.com>

¿Son eficaces los sistemas multiregistro en buscadores?

Un sistema multiregistro es un programa o aplicación web con el que podemos registrar una web en muchos buscadores introduciendo una sola vez los datos de la página, en lugar de ir buscador a buscador introduciendo los mismos datos para realizar todos los registros.

El registro en buscadores es una tarea muy importante si se quiere hacer una buena promoción de la página, puesto que no es lo mismo aparecer, entre los resultados de la búsqueda, en un puesto que 100 posiciones más abajo. Por esta razón, deberíamos ser muy concienzudos en el registro, eligiendo las palabras clave y la categoría del posible directorio donde incluir nuestro sitio, por lo menos en los buscadores más importantes, como Yahoo.

Sin embargo, existen cientos de buscadores donde queremos figurar, aunque puede que nos importe un poco menos la colocación entre los resultados... y puede que haya cientos o miles de buscadores donde queremos figurar...

Así pues, el dilema estaría entre:

- Registrar tu página a mano, asegurándote de la calidad del registro, pero utilizando mucho tiempo.
- Registrar tu página en poco tiempo, pero con una calidad menor en el registro y la colocación de la página en el directorio del buscador.

La recomendación sería hacer a mano los registros más importantes y tener una herramienta para realizar un registro masivo de la página web. Elegimos los directorios donde registrarnos a mano -de 10 a 20 buscadores- y ponemos en marcha un multiregistro para los cientos o miles que faltan.

El hecho de que en muchos buscadores haya que introducir la página en una categoría y que estas categorías varíen mucho de un buscador a otro es otra de las razones por las que un multiregistro automático puede resultar problemático. Como podéis imaginar, a los buscadores les llegan cientos de páginas cada día y si ven algunas que llegan automatizadas, donde la página no haya sido bien encajada entre las categorías, pueden desestimarla, dejando de lado su registro. Teniendo en cuenta que los sistemas multiregistro no pueden mantener al día todas las categorías de todos los buscadores, la idea de registrar la página manualmente, por lo menos en los buscadores más importantes, parece todavía más razonable.

Tipos de multiregistro y su eficacia

Interfaz web

Tenemos un [directorio donde podemos encontrar varias de estas herramientas](#) gratuitas. Son páginas web donde el creador se ha preocupado por reunir una buena cantidad de buscadores y automatizar su registro.

En DesarrolloWeb tenemos una [herramienta multiregistro en buscadores gratuita](#) y con interfaz web. El objetivo de esta herramienta es el de ofreceros la posibilidad, de dar de alta gratis vuestras páginas en una treintena de buscadores. Un servicio como éste cuesta mucho de mantener ya que los buscadores actualizan sus formularios de registro, desaparecen o ponen mecanismos para evitar que estas herramientas funcionen correctamente. Pese a ello creemos que se trata de una opción interesante para aquellos que deseen dar a conocer su página personal o no comercial sin tener que pagar por ello. Por supuesto, este tipo de servicio no ofrece ninguna garantía en lo que respecta su eficacia y, para proyectos comerciales, aconsejamos el uso de herramientas profesionales de pago.

Programas windows

También existen varios programas que se pueden comprar para realizar un registro múltiple en buscadores. La ventaja de estos programas con respecto a las versiones online con interfaz web, consiste en que detrás de los programas comerciales se encuentra una empresa que se lucra de vender su producto. Por esta razón, la empresa se preocupa mucho de mantener actualizada la base de datos de buscadores, de modo que las herramientas tengan la mayor efectividad posible.

Existen varias opciones de programas de este estilo, como [Active WebTraffic](#), [EspAñadir](#) o [SubmitWolf](#). Si bien nosotros os podemos recomendar [Active WebTraffic](#), ya que es el que utilizamos para realizar nuestros registros masivos.

Respecto al envío de altas a directorios donde el árbol de categorías es muy amplio,

donde un envío automático resulta ineficaz, como es el caso de Yahoo, Open Directory o Terra directorio, es recomendable darse de alta manualmente para elegir la categoría más adecuada al contenido de nuestra web. Por eso en la última versión del programa [Active WebTraffic](#) se ha incluido la opción de "Buscadores manuales", que facilita la inclusión manual en estos directorios donde el registro manual es importante.

Nota: para que el registro de la página sea correcto no sólo es necesario elegir la categoría adecuada, también habrá que seguir las normas del directorio para asegurarnos que el alta será aceptada por el editor que revise nuestra solicitud de alta. En el caso de Yahoo estas normas son muy estrictas y el alta manual tiene que ser perfecta para que la página sea incluida.

Al utilizar un software de multiregistro también es necesario rellenar todos los campos correctamente siguiendo los consejos que, en el caso de Active WebTraffic, se envían al propietario del software para que el alta sea eficaz. Nunca se debe utilizar el software indiscriminadamente ya que lo único que se consigue es que nuestra web sea penalizada y no aceptada por el buscador. Como regla general una página al día por dominio es suficiente, hay que ser conservador.

Si se ha introducido toda la información correctamente y se utiliza el software con prudencia, la página web será aceptada, exactamente igual a si se hubiera hecho un alta manual.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

El posicionamiento en los buscadores de Internet

Todas las empresas con cierta entidad, hoy en día disponen de un sitio web, pero muchas de ellas se plantean si esta presencia en la Red ha sido una inversión o un mero gasto de representación.

Para que nuestra presencia en Internet sea rentable es necesaria la promoción de nuestro sitio, y una de las herramientas más rentables para dar a conocer nuestro site y generar visitas son los buscadores.

Aparecer en los principales buscadores internacionales, nacionales y en los específicos de nuestro sector es principal, pero no lo es menos aparecer en un lugar destacado según ciertas palabras clave que definan nuestro negocio en la mente de nuestras audiencias. Puesto la mayoría de los internautas se conforman con los primeros resultados proporcionados por su buscador favorito.

Y para aparecer en un lugar destacado en estas útiles herramientas de búsqueda debemos conocer su funcionamiento interno. Los buscadores se dividen en dos grandes grupos: los índices y los motores de búsqueda.

Referencia: en nuestro [manual de promoción de páginas web](#) podemos encontrar gran parte de la información de este artículo, relatada con mayor detalle y más calmadamente.

Los índices dividen la información en un árbol temático de categorías y subcategorías. Aquí el ejemplo paradigmático sería Yahoo!, que nos presenta una serie de grandes categorías temáticas entre las que encontramos la subcategoría "Economía y Negocios", dentro de ella "Empresas", y esta a su vez contiene entre otras "productos y servicios para empresas" y así sucesivamente hasta ir acotando la amplitud de la categoría de sitios web, ya que no se nos permitirá proponer la inclusión de nuestro sitio web en una categoría demasiado amplia. En los índices lo

esencial es encontrar la rama ideal de este árbol temático en la cual ubicar nuestro sitio web, y digo nuestro sitio, puesto que en los índices sólo es posible incluir una página (normalmente la principal) a su directorio, aunque a menudo es posible incluirlo en dos o tres categorías. Para encontrar esta categoría ideal en la cual debería estar nuestro web, la estrategia a seguir es puramente marketiniana; ponerse en la piel de nuestro público y pensar en qué categoría nos buscará. Para ello, podemos ayudarnos de un estudio de mercado, y como no, del sentido común y de la observación de en qué categoría se encuentran ubicados nuestros principales competidores. Pero cuidado, quizá ellos no lo hayan echo tan bien y no se encuentren en la categoría ideal. Lógicamente dependiendo de la amplitud de nuestros productos o servicios, será más obvio o más difícil hallar esta categoría ideal. No obstante tras nuestra petición de alta existe un proceso de revisión humano e incluso podemos proponer una nueva categoría si no nos encontramos debidamente definidos por ninguna de las existentes.

Y sobre este proceso de revisión humana es sobre el que quiero hablar a continuación, puesto que es este el segundo factor que más diferencia a los índices de Internet de los motores de búsqueda. Cuando proponemos el alta de nuestro sitio en el índice de turno, se nos pide toda una serie de datos, como: Título de la página, URL, Definición, Ubicación geográfica, persona de contacto y correo electrónico... Y finalmente nuestro site es revisado por un surfer (un especialista en catalogar recursos) del índice que considera si nuestro site cumple con los estándares de calidad requeridos y si está bien clasificado en la categoría elegida por nosotros.

Vemos que esto es lo único que conoce el índice de nuestro sitio web; los datos suministrados en el formulario de petición de alta en el buscador. Por lo que debemos ser extremadamente cuidadosos en la definición que enviamos de nuestro site.

El caso de los motores de búsqueda es bien distinto. Podemos tomar como ejemplo a Google, y veremos que la única información que proporcionamos a un motor es la dirección URL (por ejemplo: www.miempresa.com) y quizá una dirección de correo electrónico. El resto del proceso se realiza de forma automática, ya que nuestra petición de alta en el buscador entrará en la cola de trabajo de un programa de software llamado spider (araña) que visitará la página que hemos dado de alta y a partir de ella todas las que se encuentren enlazadas y así sucesivamente. Simultáneamente nuestras páginas serán indexadas utilizando complejos algoritmos, para ser devueltas como resultado cuando un internauta utilizando el buscador, introduzca un término que se encuentre en alguna de ellas y haga una petición de extracción de información de su ingente base de datos. Vemos de esta forma que nuestro web puede aparecer en algún motor de búsqueda por la simple razón de que otra página de un tercero que está incluida en el buscador enlaza a ella en Internet.

Así, en los motores de búsqueda, para obtener una notable posición, lo esencial es el código de nuestras páginas, algo que era verdaderamente indiferente en el caso de los índices.

Teóricamente con sólo dar de alta nuestra página principal el buscador indexará todas las páginas que cuelgan de ella, pero habitualmente nos encontraremos con problemas derivados de la ventaja que se concede a las altas de pago frente a las gratuitas; el primero es el tiempo a esperar para que nuestro sitio sea introducido en la base de datos del motor de búsqueda, que puede variar entre varias semanas a varios meses según el motor en cuestión. E incluso a menudo, tras este dilatado periodo de tiempo, no seremos indexados en sus bases de datos. Y esto, en los motores que aún admiten el alta gratuita.

Como recomendación, si nuestro tiempo y energías son limitadas deberemos optar por el alta de pago en algunos buscadores.

Si conocemos las interacciones entre los distintos buscadores de Internet, descubriremos que la inclusión en alguno de ellos puede suponer la sucesiva inclusión en otros que a menudo son más "duros" con las admisiones.

Una vez que conseguimos que nuestro web aparezca en los buscadores, nuestro trabajo no habrá hecho más que empezar, ya que lo realmente valioso es aparecer en los primeros lugares por aquellas palabras clave que nuestros públicos utilizan para buscar nuestra categoría de productos, y esta sí que es una verdadera guerra, puesto que en esa lucha estamos frente a nuestros principales competidores, que también batallarán por mejorar la posición de sus páginas frente a las nuestras y las de otros competidores. Desde luego, que la complejidad dependerá de la popularidad de las palabras clave por las que queramos aparecer de forma destacada en el buscador.

Realmente cada buscador valora de distinta forma el código de nuestras páginas para ubicarla en una u otra posición de su ranking, así por ejemplo Google valora especialmente cuantas y que tipo de páginas apuntan hacia las nuestras, aplicando una lógica bastante humana, según la cual si muchos y especialmente importantes hablan de uno, es que uno es importante. Otros motores como AltaVista valoran los Meta Tags (unas líneas de código que informan al motor acerca del contenido de nuestras páginas), etc.

En general, los buscadores se fijan en la frecuencia o densidad y ubicación con la que aparecen ciertos términos en nuestras páginas, para ubicarlas en un lugar superior de sus resultados frente a otras páginas, en las cuales la frecuencia y ubicación de este término que el navegante ha introducido en la caja de búsqueda del motor aparece.

Así, una palabra que está presente en nuestra misma dirección de Internet (www.palabra.com) indica un elevado nivel de coincidencia si es el término buscado por el internauta. Después es especialmente valorada esta palabra, en el título del documento, en el primer párrafo más que en el segundo... Si está en mayúscula es más valorado, al igual que si está en negrita... Si aparece dos veces en una frase más que si aparece una, etc.

A estas alturas, seguro que a más de un lector se le ha pasado por la cabeza, la idea de llenar de términos clave la página para que aparezca en las primeras posiciones del buscador, pero lamentablemente esto ya está contemplado por estas herramientas que si encuentran demasiadas palabras repetidas o un texto de tamaño muy pequeño o con el mismo color que el fondo, etc, penalizarán nuestras páginas o incluso las eliminarán de la base de datos por tratarse de técnicas de spam (técnicas de promoción ilícitas)

Informe de **Javier Maldonado**
Director de Marketing
Mail: jmaldonado@marketalia.com
URL: <http://www.marketalia.com>



Todos los [derechos de reproducción y difusión](#) reservados a [Guiarte Multimedia S.L.](#)

CSS, HOJAS DE ESTILO PARA WEB

Introducción a las CSS

El lenguaje HTML está limitado a la hora de aplicarle forma a un documento. Esto es así porque fué concebido para otros usos (científicos sobretodo), distinto a los actuales, mucho más amplios.

Para solucionar estos problemas los diseñadores han utilizado técnicas tales como la utilización de tablas imagenes transparentes para ajustarlas, utilización de etiquetas que no son estándares del HTML y otras. Estas "trampas" han causado a menudo problemas en las páginas a la hora de su visualización en distintas plataformas.

Además, los diseñadores se han visto frustrados por la dificultad con la que, aun utilizando estos trucos, se encontraban a la hora de maquetar las páginas, ya que muchos de ellos venían maquetando páginas sobre el papel, donde el control sobre la forma del documento es absoluto.

Finalmente, otro antecedente que ha hecho necesario el desarrollo de esta tecnología consiste en que las páginas web tienen mezclado en su código HTML el contenido del documento con las etiquetas necesarias para darle forma. Esto tiene sus inconvenientes ya que la lectura del código HTML se hace pesada y difícil a la hora de buscar errores o depurar las páginas. Aunque, desde el punto de vista de la riqueza de la información y la utilidad de las páginas a la hora de almacenar su contenido, es un gran problema que estos textos estén mezclados con etiquetas incrustadas para dar forma a estos: se degrada su utilidad.

En estas páginas de CSS pretendemos dar a conocer la tecnología con un enfoque práctico para que en pocos capítulos podáis usar las CSS de una manera depurada, reflejando toda nuestra experiencia en su uso. No pretendemos explorar todos los aspectos de la tecnología ya que para realizar esto necesitaríamos un la extensión de un libro entero.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Características y ventajas de las CSS

El modo de funcionamiento de las CSS consiste en definir, mediante una sintaxis especial, la forma de presentación que le aplicaremos a:

- Un web entero, de modo que se puede definir la forma de todo el web de una sola vez.
- Un documento HTML o página, se puede definir la forma, en un pequeño trozo de código en la cabecera, a toda la página.
- Una porción del documento, aplicando estilos visibles en un trozo de la página.
- Una etiqueta en concreto, llegando incluso a poder definir varios estilos diferentes para una sola etiqueta. Esto es muy importante ya que ofrece potencia en nuestra programación. Podemos definir, por ejemplo, varios tipos de párrafos: en rojo, en azul, con márgenes, sin ellos...

La potencia de la tecnología salta a la vista. Pero no solo se queda aquí, ya que además esta sintaxis CSS permite aplicar al documento formato de modo mucho más exacto. Si antes el HTML se nos quedaba corto para maquetar las páginas y teníamos que utilizar trucos para conseguir nuestros efectos, ahora tenemos muchas más herramientas que nos permiten definir esta forma:

- Podemos definir la distancia entre líneas del documento.
- Se puede aplicar identado a las primeras líneas del párrafo.
- Podemos colocar elementos en la página con mayor precisión, y sin lugar a errores.
- Y mucho más, como definir la visibilidad de los elementos, márgenes, subrayados, tachados...

Y seguimos mostrandoo ventajas, ya que si con el HTML tan sólo podíamos definir atributos en las páginas con pixeles y porcentajes, ahora podemos definir utilizando muchas más unidades como:

- Pixels (px) y porcentaje (%), como antes.
- Pulgadas (in)
- Puntos (pt)
- Centímetros (cm)

Navegadores que lo soportan

Esta tecnología es bastante nueva, por lo que no todos los navegadores la soportan. En concreto, sólo los navegadores de Netscape versiones de la 4 en adelante y de Microsoft a partir de la versión 3 son capaces de comprender los estilos en sintaxis CSS. Además cabe destacar que no todos los navegadores implementan las mismas funciones de hojas de estilos, por ejemplo, Microsoft Internet Explorer 3 no soporta todo lo relativo a capas.

Esto quiere decir que debemos de usar esta tecnología con cuidado, ya que muchos usuarios no podrán ver los formatos que apliquemos a las páginas con CSS. Así pues, utilizad las hojas de estilos cuando estas no vayan a suponer un problema.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Usos de las CSS I

Vamos ahora a describir los diferentes usos de las CSS introducidos en el anterior capítulo. Vamos por orden, describiendo los puntos según su dificultad e importancia.

Hemos partido este capítulo en dos partes por su extensión y por haber varias formas distintas de aplicar estilos, aquí veremos las más sencillas y en el capítulo siguiente otras más complicadas pero más potentes.

CSS tiene una sintaxis propia, la veremos a través de ejemplos.

Luego se verá con detalle

Pequeñas partes de la página

Para definir estilos en secciones reducidas de una página se utiliza la etiqueta ****. Con su atributo **style** indicamos en sintaxis CSS las características de estilos. Lo vemos con un ejemplo, pondremos un párrafo en el que determinadas palabras las vamos a visualizar en color verde.

```
<p>  
Esto es un párrafo en varias palabras <SPAN style="color:green">en color verde</SPAN>. resulta  
muy fácil.  
</p>
```

Que tiene como resultado:

Esto es un párrafo con varias palabras **en color verde**. resulta muy fácil.

Estilo definido para una etiqueta

De este modo podemos hacer que toda una etiqueta muestre un estilo determinado. Por ejemplo, podemos definir un párrafo entero en color rojo y otro en color azul. Para ello utilizamos el atributo **style**, que es admitido por todas las etiquetas del HTML (siempre y cuando dispongamos de un navegador compatible con CSS).

```
<p style="color:#990000">
Esto es un párrafo de color rojo.
</p>
<p style="color:#000099">
Esto es un párrafo de color azul.
</p>
```

Que tiene como resultado:

Esto es un párrafo de color rojo.

Esto es un párrafo de color azul.

Estilo definido en una parte de la página

Con la etiqueta **<DIV>** podemos definir secciones de una página y aplicarle estilos con el atributo **style**, es decir, podemos definir estilos de una vez a todo un bloque de la página.

```
<div style="color:#000099; font-weight:bold">
<h3>Estas etiquetas van en <i>azul y negrita</i></h3>
<p>
Seguimos dentro del DIV, luego permanecen los etilos
</p>
</div>
```

Que tiene como resultado:

Estas etiquetas van en azul y negrita

Seguimos dentro del DIV, luego permanecen los etilos

Hasta aquí este capítulo, en el siguiente seguiremos viendo formas más avanzadas de usar las CSS.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Usos de las CSS y II

Estilo definido para toda una página

Podemos definir, en la cabecera del documento, estilos para que sean aplicados a toda la página. Es una manera muy cómoda de darle forma al documento y muy potente, ya que estos estilos serán seguidos en toda la página y nos ahorraremos así muchas etiquetas HTML que apliquen forma al documento. Además, si deseamos cambiar los estilos de la página lo haremos de una sola vez.

Este ejemplo es más complicado, puesto que se utiliza la sintaxis CSS de manera más avanzada. Pero no te preocupes puesto que con los ejemplos irás aprendiendo su uso y más tarde comentaremos la sintaxis en profundidad.

En el ejemplo vemos que se utiliza la etiqueta <STYLE> colocada en la cabecera de la página para definir los distintos estilos del documento.

A grandes rasgos, entre de <STYLE> y </STYLE>, se coloca el nombre de la etiqueta que queremos definir los estilos y entre llaves -{}- colocamos en sintaxis CSS las características de estilos.

```
<html>
<head>
<title>Ejemplo de estilos para toda una página</title>
<STYLE type="text/css">
<!--
H1 {text-decoration: underline; text-align:center}
P {font-Family:arial,verdana; color: white; background-color: black}
BODY {color:black;background-color: #cccccc; text-indent:1cm}
// -->
</STYLE>
</head>

<body>
<h1>Página con estilos</h1>
Bienvenidos...
<p>Siento ser tan hortera, pero esto es un ejemplo sin mucha importancia</p>
</body>
</html>
```

Como se puede apreciar en el código, hemos definido que la etiqueta <H1> se presentará

- Subrayado
- Centrada

También, por ejemplo, hemos definido que el cuerpo entero de la página (etiqueta <BODY>) se le apliquen los estilos siguientes:

- Color del texto negro
- Color del fondo grisáceo
- Margen lateral de 1 centímetro

Caber destacar que si aplicamos estilos a la etiqueta <BODY>, estos serán heredados por el resto de las etiquetas del documento. Esto es así siempre y cuando no se vuelvan a definir esos estilos en las siguientes etiquetas, en cuyo caso el estilo de la etiqueta más concreta será el que mande. Puede verse este detalle en la etiqueta <P>, que tiene definidos estilos que ya fueron definidos para <BODY>. Los estilos que se tienen en cuenta son los de la etiqueta <P>, que es más concreta.

Por último, ha de apreciarse los comentarios HTML que engloban toda la declaración de estilos: <!--Declaración de estilos-->. Estos comentarios se utilizan para que los navegadores antiguos, que no comprenden la sintaxis CSS, no incluyan ese texto en el cuerpo de la página. Si no se pusiera, los navegadores antiguos (por ejemplo Netscape 3) escribirían ese "feo código" en la página.

[Pulsa para ver el ejemplo anterior.](#)

Hemos preparado la misma página, pero con declaraciones de estilos distintas, para que comprobéis las diferencias en la forma del documento con sólo unos cambios en sus estilos. [Puedes verla pinchando aquí.](#)

Estilo definido para todo un sitio web

Una de las características más potentes de la programación con hojas de estilos consiste en que, de una vez, podemos definir los estilos de todo un sitio web. Esto se consigue creando un archivo donde tan sólo colocamos las declaraciones de estilos de la página y enlazando todas las páginas del sitio con ese archivo. De este modo, todas las páginas comparten una misma declaración de estilos y, por tanto, si la cambiamos, cambiarán todas las páginas. Con las ventajas añadidas de que se ahorra en líneas de código HTML (lo que reduce el peso del documento) y se evita la molestia de definir una y otra vez los estilos con el HTML, tal como se comentó anteriormente.

Veamos ahora cómo el proceso para incluir estilos con un fichero externo.

1- Creamos el fichero con la declaración de estilos

Es un fichero de texto normal, que puede tener cualquier extensión, aunque le podemos asignar la extensión .css para aclararnos qué tipo de archivo es. El texto que debemos incluir debe ser escrito exclusivamente en sintaxis CSS, es decir, sería erróneo incluir código HTML en él: etiquetas y demás. Podemos ver un ejemplo a continuación.

```
P {
font-size : 12pt;
font-family : arial,Helvetica;
font-weight : normal;
}

H1 {
font-size : 36pt;
font-family : verdana,arial;
text-decoration : underline;
text-align : center;
background-color : Teal;
}

TD {
font-size : 10pt;
font-family : verdana,arial;
text-align : center;
background-color : 666666;
}

BODY {
background-color : #006600;
font-family : arial;
color : White;
}
```

2- Enlazamos la página web con la hoja de estilos

Para ello, vamos a colocar la etiqueta <LINK> con los atributos

- **rel="STYLESHEET"** indicando que el enlace es con una hoja de estilos
- **type="text/css"** porque el archivo es de texto, en sintaxis CSS
- **href="estilos.css"** indica el nombre del fichero fuente de los estilos

Veamos una página web entera que enlaza con la declaración de estilos anterior:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">

<html>
<head>
<link rel="STYLESHEET" type="text/css" href="estilos.css">
<title>P&acute;gina que lee estilos</title>
</head>

<body>
<h1>P&acute;gina que lee estilos</h1>
Esta p&acute;gina tiene en la cabecera la etiqueta necesaria para enlazar con la hoja de estilos. Es
muy f&acute;cil.
<br>
<br>
<table width="300" cellspacing="2" cellpadding="2" border="0">
<tr>
<td>Esto est&acute; dentro de un TD, luego tiene estilo propio, declarado en el fichero
externo</td>
</tr>
<tr>
<td>La segunda fila del TD</td>
</tr>
</table>

</body>
</html>
```

[El resultado conseguido se puede ver aquí](#)

Reglas de importancia en los estilos

Los estilos se heredan de una etiqueta a otra, como se indicó anteriormente. Por ejemplo, si tenemos declarado en el <BODY> unos estilos, por lo general, estas declaraciones también afectarán a etiquetas que estén dentro de esta etiqueta, o lo que es lo mismo, dentro de todo el cuerpo.

En muchas ocasiones más de una declaración de estilos afecta a la misma porción de la página. Siempre se tiene en cuenta la declaración más particular. Pero las declaraciones de estilos se pueden realizar de múltiples modos y con varias etiquetas, también entre estos modos hay una jerarquía de importancia para resolver conflictos entre varias declaraciones de estilos distintas para una misma porción de página. Se puede ver a continuación esta jerarquía, primero ponemos las formas de declaración más generales, y por tanto menos respetadas en caso de conflicto:

- Declaración de estilos con fichero externo. (Para todo un sitio web)
- Declaración de estilos para toda la página. (Con la etiqueta <STYLE> en la cabecera de la página)
- Estilos definidos en una parte de la página. (Con la etiqueta <DIV>)
- Definidos en una etiqueta en concreto. (Utilizando el atributo style en la etiqueta en cuestión)
- Declaración de estilo para una porción pequeña del documento. (Con la etiqueta)

Ya vimos cómo incluir estilos en la página, de todas las maneras posibles e hicimos un repaso con la lista anterior. Ahora estás en condiciones de empezar a usar las hojas de estilo en cascada para mejorar tus páginas y aumentar la productividad de tu trabajo. Pero estate atento a los siguientes capítulos donde aprenderás las

lecciones que te faltan para dominar bien la materia: conocer la sintaxis, los distintos atributos de estilos y otras cosas que mejorarán tus páginas.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Otra manera de definir estilos en un archivo externo

Veamos ahora otra manera de importar una declaración externa de estilos CSS: `@import url("archivo_a_importar.css")`, que se utiliza para importar unas declaraciones de estilos guardadas en la ruta que se indica entre paréntesis. (las comillas son opcionales, pero los paréntesis son obligatorios, por lo menos, en Explorer).

Se debe incluir en la declaración de estilos global a una página, es decir entre las etiquetas `<style type="text/css">` y `</style>`, que se colocan en la cabecera del documento.

Es importante señalar que la sentencia de importación del archivo CSS se debe escribir en la primera línea de la declaración de estilos, algo parecido al código siguiente.

```
<style type="text/css">
@import url ("estilo.css");
body{
  background-color:#ffffcc;
}
</style>
```

El funcionamiento es el mismo que si escribiésemos todo el fichero a importar dentro de las etiquetas de los estilos, con la salvedad de que, si redefinimos dentro del código HTML (entre las etiquetas `</style>`) estilos que habían quedado definidos en el archivo externo, los que se aplicarán serán los que hayamos redefinido.

Así, en el ejemplo anterior, aunque hubiésemos definido en `estilo.css` un color de fondo para la página, el color que prevalecería sería el definido a continuación de la importación: `#ffffcc`

La diferencia entre este tipo de importación del tipo y la que hemos visto anteriormente:

```
<link rel="stylesheet" type="text/css" href="hoja.css">
```

Es que `@import url ("estilo.css")` se suele utilizar cuando hay unas pautas básicas en el trabajo con los estilos (que se definen en un archivo a importar) y unos estilos específicos para cada página, que se definen a continuación, dentro del código HTML entre las etiquetas `</style>`, como es el caso del ejemplo visto anteriormente.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Sintaxis CSS

Tal como se vio en los ejemplos la sintaxis es bastante sencilla y repetitiva. Vamos a verla:

- Para definir un estilo se utilizan atributos como `font-size`, `text-decoration`... seguidos de dos puntos y el valor que le deseemos asignar. Podemos definir un estilo a base de definir muchos atributos separados por punto y coma.
Ejemplo:
`font-size: 10pt; text-decoration: underline; color: black;` (el último punto y coma de la lista de atributos es opcional)
- Para definir el estilo de una etiqueta se escribe la etiqueta seguida de la lista de atributos encerrados entre llaves.
Ejemplo:
`H1{text-align: center; color:black}`
- Los valores que se pueden asignar a los atributos de estilo se pueden ver en una tabla en el siguiente capítulo. Muchos estos valores son unidades de medida, por ejemplo, el valor del tamaño de un margen o el tamaño de la fuente. Las unidades de medida son las siguientes:

Puntos	pt
Pulgadas	in
Centímetros	cm
pixels	px

Hasta aquí, queda dicho todo lo relativo a la sintaxis. En el siguiente capítulo podrás encontrar una lista de los atributos de las hojas de estilo en cascada.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Atributos de las hojas de estilo

Tanto para practicar en tu aprendizaje como para trabajar con las CSS lo mejor es disponer de una tabla donde se vean los distintos atributos y valores de estilos que podemos aplicarle a las páginas web.

Aquí puedes ver la tabla de los atributos CSS, tenla a mano cuando utilices las CSS.

Nombre del atributo	Posibles valores	Ejemplos
FUENTES - FONT		
color	valor RGB o nombre de color	color: #009900; color: red;
Sirve para indicar el color del texto. Lo admiten casi todas las etiquetas de HTML. No todos los nombres de colores son admitidos en el estandar, es aconsejable entonces utilizar el valor RGB.		
font-size	xx-small x-small small medium large x-large xx-large Unidades de CSS	font-size:12pt; font-size: x-large;
Sirve para indicar el tamaño de las fuentes de manera más rígida y con mayor exactitud.		
font-family	serif sans-serif cursive fantasy monospace	font-family:arial,Helvetica; font-size: fantasy;

Todas las fuentes habituales

Con este atributo indicamos la familia de tipografía del texto. Los primeros valores son genéricos, es decir, los exploradores los comprenden y utilizan las fuentes que el usuario tenga en su sistema.

También se pueden definir con tipografías normales, como ocurría en html. Si el nombre de una fuente tiene espacios se utilizan comillas para que se entienda bien.

font-weight	normal bold bolder lighter 100 200 300 400 500 600 700 800 900	font-weight:bold; font-weight: 200;
--------------------	---	--

Sirve para definir la anchura de los caracteres, o dicho de otra manera, para poner negrillas con total libertad.

Normal y 400 son el mismo valor, así como bold y 700.

font-style	normal italic oblique	font-style:normal; font-style: italic;
-------------------	---------------------------	---

Es el estilo de la fuente, que puede ser normal, itálica u oblicua. El estilo oblique es similar al italic.

PÁRRAFOS - TEXT

line-height	normal y unidades CSS	line-height: 12px; line-height: normal;
--------------------	-----------------------	--

El alto de una línea, y por tanto, el espaciado entre líneas. Es una de esas características que no se podían modificar utilizando HTML.

text-decoration	none [underline overline line-through]	text-decoration: none; text-decoration: underline;
------------------------	---	---

Para establecer la decoración de un texto, es decir, si está subrayado, sobrerayado o tachado.

text-align	left right center justify	text-align: right; text-align: center;
-------------------	---------------------------------	---

Sirve para indicar la alineación del texto. Es interesante destacar que las hojas de estilo permiten el justificado de texto, aunque recuerda que no tiene por que funcionar en todos los sistemas.

text-indent	Unidades CSS	text-indent: 10px; text-indent: 2in;
--------------------	--------------	---

Un atributo que sirve para hacer sangrado o márgenes en las páginas. Muy útil y novedosa.

text-transform	capitalize uppercase lowercase none	text-transform: none; text-transform: capitalize;
-----------------------	--	--

Nos permite transformar el texto, haciendo que tenga la primera letra en mayúsculas de todas las palabras, todo en mayúsculas o minúsculas.

FONDO - BACKGROUND

Background-color	Un color, con su nombre o su valor RGB	background-color: green; background-color: #000055;
-------------------------	---	--

Sirve para indicar el color de fondo de un elemento de la página.

Background-image	El nombre de la imagen con su camino relativo o absoluto	background-image: url(mármol.gif) ; background-image: url(http://www.x.com/fondo.gif)
-------------------------	--	---

Colocamos con este atributo una imagen de fondo en cualquier elemento de la página, se puede ver una [página de ejemplo](#)

BOX - CAJA

Margin-left	Unidades CSS	margin-left: 1cm; margin-left: 0,5in;
--------------------	--------------	--

Indicamos con este atributo el tamaño del margen a la izquierda

Margin-right	Unidades CSS	margin-right: 5%; margin-right: 1in;
---------------------	--------------	---

Se utiliza para definir el tamaño del margen a la derecha

Margin-top	Unidades CSS	margin-top: 0px; margin-top: 10px;
-------------------	--------------	---------------------------------------

Indicamos con este atributo el tamaño del margen arriba de la página

Margin-bottom	Unidades CSS	margin-bottom: 0pt; margin-top: 1px;
----------------------	--------------	---

Con el se indica el tamaño del margen en la parte de abajo de la página

Padding-left	Unidades CSS	padding-left: 0.5in; padding-left: 1px;
---------------------	--------------	--

Indica el espacio insertado, por la izquierda, entre el borde del elemento-continente y el contenido de este. Es parecido a el atributo cellpadding de las tablas.

El espacio insertado tiene el mismo fondo que el fondo del elemento-continente.

Padding-right	Unidades CSS	padding-right: 0.5cm; padding-right: 1pt;
----------------------	--------------	--

Indica el espacio insertado, en este caso por la derecha, entre el borde del elemento-contenedor y el contenido de este. Es parecido a el atributo cellpadding de las tablas. El espacio insertado tiene el mismo fondo que el fondo del elemento-contenedor.

Padding-top	Unidades CSS	padding-top: 10pt; padding-top: 5px;
--------------------	--------------	---

Indica el espacio insertado, por arriba, entre el borde del elemento-contenedor y el contenido de este.

Padding-bottom	Unidades CSS	padding-right: 0.5cm; padding-right: 1pt;
-----------------------	--------------	--

Indica el espacio insertado, en este caso por abajo, entre el borde del elemento-contenedor y el contenido de este.

Border-color	color RGB y nombre de color	border-color: red; border-color: #ffccff;
---------------------	-----------------------------	--

Para indicar el color del borde del elemento de la página al que se lo aplicamos. Se puede poner colores por separado con los atributos border-top-color, border-right-color, border-bottom-color, border-left-color.

Border-style	none dotted solid double groove ridge inset outset	border-style: solid; border-style: double;
---------------------	--	---

El estilo del borde, los valores significan: none=ningun borde, dotted=punteado (no parece funcionar), solid=sólido, double=doble borde, y desde groove hasta outset son bordes con varios efectos 3D.

border-width	Unidades CSS	border-width: 10px; border-width: 0.5in;
---------------------	--------------	---

El tamaño del borde del elemento al que lo aplicamos.

Para ver otros ejemplos de Box [pulsar aquí](#)

float	none left right	float: right;
--------------	---------------------	---------------

Sirve para alinear un elemento a la izquierda o la derecha haciendo que el texto se agrupe alrededor de dicho elemento. Igual que el atributo align en imágenes en sus valores right y left.

clear	none right left	clear: right;
--------------	---------------------	---------------

Si este elemento tiene a su altura imágenes u otros elementos alineados a la derecha o la izquierda, con el atributo clear hacemos que se coloque en un lugar donde ya no tenga esos elementos a el lado que indiquemos.

Para ver una página que utiliza float y clear [pulsar aquí](#)

La especificación de estilos CSS es muy amplia, seguro que se queda en el tintero algún atributo de estilo, pero creo que la inmensa mayoría están, y por supuesto la selección de los más importantes.

[Textos de este capítulo en](#)  [css6.zip 5,4Kb](#)

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Trucos avanzados con CSS

Las hojas de estilos son un tema largo del que se han escrito libros enteros. Nosotros nos centramos en los temas prácticos y por ello vamos a acabar ya con este capítulo, en unos cuantos puntos

Definir estilos utilizando clases

Las clases nos sirven para crear definiciones de estilos que se pueden utilizar repetidas veces.

Una clase se puede definir entre las etiquetas <STYLE> (en la cabecera del documento), o en un archivo externo a la página. Para definir las utilizamos la

siguiente sintaxis, un punto seguido del nombre de la clase y entre llaves los atributos de estilos deseados. De esta manera:

```
.nombredelaclase {atributo: valor;atributo2:valor2; ...}
```

Una vez tenemos una clase, podemos utilizarla en cualquier etiqueta HTML. Para ello utilizaremos el atributo class, poniéndole como valor el nombre de la clase, de esta forma:

```
<ETIQUETA class="nombredelaclase">
```

Ejemplo de la utilización de clases

```
<html>
<head>
<title>Ejemplo de la utilizaci&ocaron de clases</title>
<STYLE type="text/css">
.fondonegroletrasblancas {background-color:black;color:white;font-size:12;font-family:arial}
.letrasverdes {color:#009900}
</STYLE>
</head>

<body>
<h1 class=letrasverdes>Titulo 1</h1>
<h1 class=fondonegroletrasblancas>Titulo 2</h1>

<p class=letrasverdes>
Esto es un p&acirc;rafo con estilo de letras verdes</p>
<p class=fondonegroletrasblancas>
Esto es un p&acirc;rafo con estilo de fondo negro y las letras blancas. Es todo!</p>
</body>
</html>
```

[Ver el ejemplo anterior](#)

Estilo en los enlaces

Una técnica muy habitual, que se puede realizar utilizando las hojas de estilo en cascada y no se podía en HTML, es la definición de estilos en los enlaces, quitándoles el subrayado o hacer enlaces en la misma página con distintos colores.

Para aplicar estilo a los enlaces debemos definirlos para los distintos tipos de enlaces que existen (visitados, activos...). Utilizaremos la siguiente sintaxis, en la declaración de estilos global de la página (<STYLE>) o del sitio (Definición en un archivo externo):

Enlaces normales

```
A:link {atributos}
```

Enlaces visitados

```
A:visited {atributos}
```

Enlaces activos (Los enlaces están activos en el preciso momento en que se pincha sobre ellos)

```
A:active {atributos}
```

Enlaces hover (Cuando el ratón está encima de ellos, solo funciona en ieplorer)

```
A:hover {atributos}
```

El atributo para definir enlaces sin subrayado es **text-decoration:none**, y para darles color es color:tu_color.

También podemos definir el estilo de cada enlace en la propia etiqueta <A>, con el atributo style. De esta manera podemos hacer que determinados enlaces de la página se vean de manera distinta

Ejemplo de estilos en enlaces

```
<html>
<head>
<title>Ejemplos de estilo en enlaces</title>
<STYLE type="text/css">
A:link {text-decoration:none;color:#0000cc;}
A:visited {text-decoration:none;color:#ffcc33;}
A:active {text-decoration:none;color:#ff0000;}
A:hover {text-decoration:underline;color:#999999;font-weight:bold}
</STYLE>
</head>

<body>

<a href="http://dominioinexistente.nofunciona.com">Enlace normal</a>
<br>
<br>
<a href="enlaces.html">Enlace visitado</a>
Pulsar este enlace para verlo activo,
poner el ratón por encima para que cambie.

</body>
</html>
```

[Ver el ejemplo anterior](#)

URL como valor de un atributo:

Determinados atributos de estilos, como **background-image** necesitan una URL como valor, para indicarlas podemos definir tanto caminos relativos como absolutos. Así pues, podemos indicar la URL de la imagen de fondo de estas dos maneras:

background-image: url(fondo.gif) En caso de que la imagen esté en el mismo directorio que la página. **background-image: url(http://www.desarrolloweb.com/images/fondo.gif)** ¶

Ocultar estilos en navegadores antiguos

En caso de definir dentro de la etiqueta <STYLE> unas declaraciones de estilos debemos asegurarnos que estas no se imprimirán en la página web con navegadores antiguos. Pensar en un navegador que no reconozca la etiqueta <STYLE>, pensará que corresponde con algo que no entiende y se olvidará de la etiqueta. Lo siguiente que encuentra es texto normal y hará que este se vea en la página, como con cualquier otro texto.

Para evitarlo debemos ocultar con comentarios HTML (<!-- esto es un comentario -->) todo lo que hay dentro de la etiqueta <STYLE>. Se puede ver un ejemplo de esto a continuación:

De este modo hemos terminado nuestro manual de CSS, que espero pueda ayudar a hacer páginas mejores y más rápidamente.

Quiero recordaros que siempre es útil ver como han hecho sus páginas otros programadores de Internet. Para ver una página definida enteramente con hojas de estilos visitar la dirección www.guiarte.com

Para ver otras páginas que enseñan a utilizar las hojas de estilos visitar la [sección correspondiente a CSS de nuestro buscador](#):

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Aplicación de estilo avanzada a los enlaces

En este artículo vamos a ver cómo podríamos crear una barra de navegación bastante dinámica utilizando únicamente las Hojas de Estilo en Cascada. En el ejemplo vamos a construir una barra de navegación que contiene enlaces de varios colores que cambian de tonalidad al pasar el ratón por encima.

Podemos [ver el resultado de nuestra barra pulsando este enlace](#), para tener una idea más exacta de lo que queremos conseguir.

Cómo poner estilo a los enlaces

Ya lo vimos en capítulos anteriores de nuestro [manual de CSS](#), pero lo repetimos aquí. Se define el estilo de los enlaces asignando su apariencia en sus distintos estados:

- Enlace no visitado. Se define con el atributo link.
- Enlace visitado. Se define con el atributo visited.
- Enlace activo (cuando se está pulsando). Se define con active.
- Enlace con el reton encima. Se define con hover.

Esta definición se realiza en la cabecera de la página, entre las etiquetas <STYLE> Y </STYLE>, y es global a toda la página.

Un ejemplo de esto se puede ver en esta declaración de estilos:

```
<STYLE type="text/css">
  A:link {text-decoration:none;color:#0000cc;}
  A:visited {text-decoration:none;color:#ffcc33;}
  A:active {text-decoration:none;color:#ff0000;}
  A:hover {text-decoration:underline;color:#999999;}
</STYLE>
```

Cómo dar estilo a un enlace en concreto

Si queremos resaltar nuestra barra de navegación probablemente querramos colocarla en una tabla de nuestra página web, con un color que contraste un poco con el fondo. En un caso como este, será necesario que los enlaces de la barra de navegación y los enlaces normales de la página tengan colores distintos, por estar situados sobre dos tipos de fondos distintos.

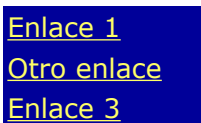
Es por esto que los enlaces de la barra van a tener un color distinto de los definidos en la cabecera de la página, con los estilos. Esto lo podemos conseguir de esta manera.

```
<a href="#" style="color:#ff0000">Mi enlace</a>
```

Hemos definido el color de un enlace de una manera específica, utilizando el atributo style, de modo que este enlace siempre tendrá el color indicado,

independientemente de su estado.

Es un enlace amarillo, que quedaría muy bien resaltado sobre un fondo oscuro, como se puede ver en el ejemplo de barra de navegación siguiente.



En la tabla anterior tenemos enlaces amarillos en una web donde los enlaces son azules por defecto.

Cómo utilizar las clases al aplicar estilo a los enlaces

También vimos en anteriores capítulos que el uso de clases puede ser muy útil a la hora de definir estilos especiales que podemos aplicar a las etiquetas que queramos. A la hora de trabajar con los enlaces también podemos aplicar las clases para definir distintos tipos de enlaces, que tienen distintos tipos de estilos.

```
A.clase1:visited {color:#ff0000;}
A.clase1:active {color:#ff0000;}
A.clase1:link {color:#ff0000}
A.clase1:hover {color:#00ff00}
```

La ventaja al utilizar las clases con los estilos de los enlaces es que podemos especificar un formato distinto al enlace dependiendo de su estado: visitado o no, activo o con el ratón sobre él.

Por si no quedó claro, al especificar el estilo con el atributo style del enlace sólo podíamos decir que el enlace lo queremos en amarillo, y siempre lo tendremos en amarillo (sea visitado o no, activo, o estemos o no con el ratón encima). Con las clases definimos un nuevo tipo de enlace al que podemos dar distintos formatos dependiendo su estado.

Otras ventajas de utilizar las clases consisten en que escribimos una única vez los estilos y que podemos cambiar el color de todos los enlaces de la clase con cambiar la declaración.

A partir de lo que acabamos de aprender podemos crear el ejemplo de barra de navegación dinámica utilizando CSS que habíamos visto al principio del capítulo. El código sería el siguiente.

```
<html>
<head>
  <title>Ejemplo CSS para enlaces</title>
  <style type="text/css">
    A:link {color:#0000cc;}
    A:visited {color:#0000cc;}
    A:active {color:#0000cc;}
    A:hover {color:#0000ff;}

    A.clase1:visited {color:#ffff00;}
    A.clase1:active {color:#ffff00;}
    A.clase1:link {color:#ffff00}
    A.clase1:hover {color:#00ff00}

    A.clase2:visited {font-size:12;color:#ffffff;}
    A.clase2:active {font-size:12;color:#ffffff;}
    A.clase2:link {font-size:12;color:#ffffff;}
    A.clase2:hover {font-size:12;color:#fff33;}

    body {font-family:arial;font-size:11;font-weight:bold}
    td {font-family:arial;font-size:11;font-weight:bold}
```

```

</style>
</head>

<body>
<a href="#">Este enlace es normal</a>
<br>
<br>
<br>
<span style="font-weight:normal;font-size:10">
Los enlaces de esta barra son especiales,
<br>
están definidos por clases
</span>
<br>
<table width="110" cellspacing="1" cellpadding="2" border="0">
<tr>
<td bgcolor="#aa0000"><a href="#" class="clase2">Opciones 1</a></td>
</tr>
<tr>
<td bgcolor="red"><a href="#" class="clase1">Enlace clase1</a></td>
</tr>
<tr>
<td bgcolor="red"><a href="#" class="clase1">Otro de clase1</a></td>
</tr>
<tr>
<td bgcolor="red"><a href="#" class="clase1">Más enlaces</a></td>
</tr>
<tr>
<td bgcolor="red"><a href="#" class="clase1">Todavía más</a></td>
</tr>
</table>

</body>
</html>

```

Se puede [ver la página en funcionamiento en este enlace](#). Esperamos que sirva de provecho este pequeño taller de CSS para vuestras creaciones.

Quiero agradecer la ayuda prestada para la elaboración de este artículo a Rafael Chacón, que nos mandó un generoso mail con el truquillo que hemos expuesto aquí.

Informe de **Miguel Angel Alvarez**
 Director desarrolloweb.com
 Mail: eugim@desarrolloweb.com

Qué son las capas

Veamos una pequeña introducción a lo que son las capas, la etiqueta HTML <DIV> utilizada para construirla y los atributos CSS con los que podemos aplicar estilos.

Como ya hemos visto en nuestro [manual de CSS](#), sirve para aplicarle estilo a una pequeña parte de una página HTML. Por ejemplo, con ella podríamos hacer que una parte de un párrafo se colorea en rojo. Con no es habitual englobar un trozo muy grande de texto, por ejemplo el que comprenda a varios párrafos.

Con <DIV> también podemos aplicar estilo a partes de la página HTML.

La diferencia entre y <DIV> es que con esta última si que podemos aplicar estilo a una parte más amplia de la página, por ejemplo a tres párrafos.

Además que la etiqueta <DIV> tiene un uso adicional que es el de crear divisiones en la página a las que podremos aplicar una cantidad adicional de atributos para modificar sus comportamientos. Por ejemplo, con el atributo align de HTML podemos alinear la división al centro, izquierda, derecha o justificado. Pero **su uso más destacado es el de convertir esa división en una capa.**

Una capa es una división, una parte de la página, **que tiene un comportamiento muy independiente** dentro de la ventana del navegador, ya que la podemos colocar en cualquier parte de la misma y la podremos mover por ella independientemente, por poner dos ejemplos. En el uso de capas se basan muchos de los efectos más comunes del DHTML.

Las etiquetas <LAYER> e <ILAYER> tienen como objetivo construir capas, pero estas no son compatibles más que con Netscape, por lo que es recomendable utilizar la etiqueta <DIV> para hacer capas preferentemente a las otras dos.

Los atributos que podemos aplicar a estas etiquetas, pero en concreto a las dos recomendadas y <DIV>, son principalmente de estilos CSS. Estos atributos nos permiten, como hemos podido ver en el manual de hojas de estilo en cascada de desarrolloweb, modificar de una manera muy exhaustiva la presentación de los contenidos en la página. Para aplicar estilos a estas etiquetas se utiliza el atributo de HTML style, de esta manera:

```
<SPAN style="text-decoration:underline;font-weight:bold">...</SPAN>
```

```
<DIV style="color:red;font-size:10px">...</DIV>
```

Como ya pudimos ver muchos ejemplos en el [manual de CSS](#), nos referimos a él para ampliar esta información. Pero no habíamos visto todavía una serie de atributos que nos sirven para posicionar la división en la página como una capa. Estos atributos se pueden aplicar a la etiqueta <DIV> que es la que servía para crear capas compatibles con todos los navegadores.

Los atributos para que la división sea una capa son varios y se pueden ver a continuación.

```
<div id="c1" style="position:absolute; left: 200px; top: 100px;">
Hola!
</div>
```

El primero, **position**, indica que se posicione de manera absoluta en la página y los segundos, **left** y **top**, son la distancia desde el borde izquierdo de la página y el borde superior.

Hay otros atributos especiales para capas como **width** y **height** para indicar la anchura y altura de la capa, **Z-index** que sirve para indicar qué capas se ven encima de qué otras, **clip** que sirve para recortar una capa y hacer que partes de ella no sean visibles, o **visibility** para definir si la capa es visible o no. Estos y otros atributos los veremos en el [siguiente capítulo, donde hablaremos del posicionamiento de capas](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Atributos para capas

Hemos visto en el capítulo anterior [qué son las capas y algunas pequeñas muestras sobre cómo crearlas y darle algún estilo](#). Ahora vamos a ver en detenimiento los atributos específicos para aplicar posicionamiento a una capa y otros estilos.

Antes que nada cabe decir que una capa puede tener cualquier atributo de estilos de los que hemos visto en el [manual de CSS](#). Así, el atributo color indica el color del texto de la capa, el atributo font-size indica el tamaño del texto y así con todos los atributos ya vistos.

Ahora bien, existen una serie de atributos que sirven para indicar la forma, el tamaño de las capas, la visibilidad, etc, que no hemos visto en capítulos anteriores y que veremos a continuación.

Atributo position

Indica el tipo de posicionamiento de la capa. Puede tener dos valores, relative o absolute.

- relative indica que la posición de la capa es relativa a el lugar donde se estaba escribiendo en la página en el momento de escribir la capa con su etiqueta

- absolute indica que la posición de la capa se calcula con respecto al punto superior izquierdo de la página.

Atributo top

Indica la distancia en vertical donde se colocará la capa. Si el atributo position es absolute, top indica la distancia del borde superior de la capa con respecto al borde superior de la página. Si el atributo position era relative, top indica la distancia desde donde se estaba escribiendo en ese momento en la página hasta el borde superior de la capa.

Atributo left

Básicamente funciona igual que el atributo top, con la diferencia que el atributo left indica la distancia en horizontal a la que estará situada la capa.

Atributo height

Sirve para indicar el tamaño de la capa en vertical, es decir, su altura.

Atributo width

Indica la anchura de la capa

Atributo visibility

Sirve para indicar si la capa se puede ver en la página o permanece oculta al usuario. Este atributo puede tener tres valores.

- visible sirve para indicar que la capa se podrá ver.
- hidden indicará que la capa está oculta.

- inherit es el valor por defecto, que quiere decir que hereda la visibilidad de la capa donde está metida la capa en cuestión. Si la capa no está metida dentro de ninguna otra se supone que está metida en la capa documento, que es toda la página y que siempre está visible.

Atributo z-index

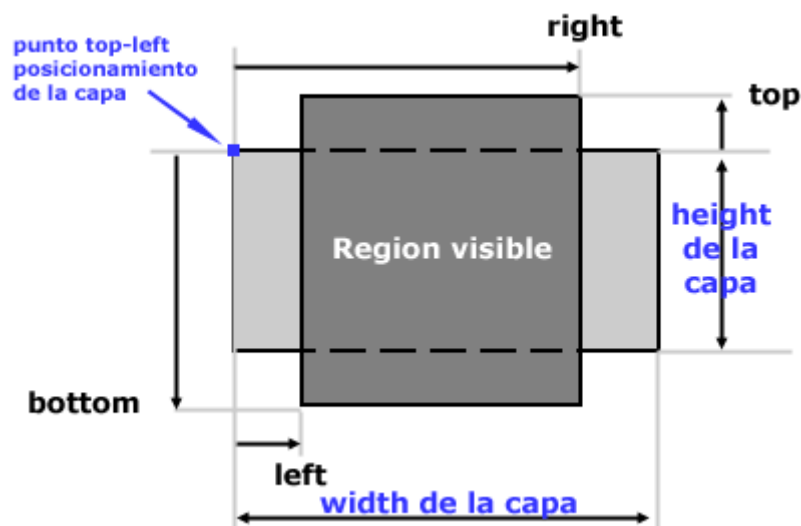
Sirve para indicar la posición sobre el eje z que tendrán las distintas capas de la página. Dicho de otra forma, con z-index podemos decir qué capas se verán encima o debajo de otras, en caso de que estén superpuestas. El atributo z-index toma valores numéricos y a mayor z-index, más arriba se verá la página.

Atributo clip

Es un atributo un poco difícil de explicar. En concreto sirve para recortar determinadas áreas de la capa y que no se puedan ver. Una capa que está visible se puede recortar para que se vea, pero que no se vea algún trozo de esta. El clipping se indica por medio de 4 valores, con esta sintaxis.

rect (<top>, <right>, <bottom>, <left>)

Los valores <top>, <right>, <bottom> y <left> indican distancias que se pueden apreciar en este esquema.



Este es un ejemplo de capa que utiliza todos los atributos que hemos visto en este artículo y alguno más para aplicar estilo a la capa.

```
<div style="clip: rect(0,158,148,15); height: 250px; width: 170px; left: 10px; top: 220px; position: absolute; visibility: visible; z-index:10; font-size: 14pt; font-family: verdana; text-align: center; background-color: #bbbbbb">
```

Esta capa tiene un clipping, por eso se ve entrecortada.

Esto es una capa de prueba

</div>

Puede [verse el ejemplo en una página web](#), donde también podrá apreciarse el efecto conseguido al realizar el clipping.

Informe de **Miguel Angel Alvarez**

Problema con el posicionamiento absoluto de capas

He recibido una consulta en mi correo sobre colocación de capas de manera absoluta, pero en la que no nos importe la definición de la pantalla del usuario y otros ir y venir de los elementos HTML. Nuestro compañero expresó su duda de la siguiente manera:

Si trabajamos con position:absolute dando un left y un top funciona si tienes tu página alineada a la izquierda. Mi página está alineada en el centro, entonces lo que sucede es que dependiendo de la resolución de pantalla que tengas (ancho de 800px,1024px,etc) me baila toda la página y no cuadra nada.

Primero que todo, debemos saber que si trabajamos con el position relative las capas se colocan en el lugar donde aparecen dentro del código HTML. De este modo, si colocamos una capa con position relative dentro de una celda de una tabla, dicha capa aparecería dentro de la celda donde la estamos colocando, independientemente del lugar donde se sitúe la celda al cambiar la definición de la pantalla.

El problema de esta solución es que la capa haría crecer la celda de la tabla donde queremos colocarla (al igual que cualquier otro elemento HTML que colocásemos dentro de la tabla) y es muy probable que nuestro diseño no nos permita este hecho. Seguramente ya habrías notado este problema y si no es así te invito a que crees la capa que intentas colocar con el atributo position a relative para ver si con eso tu problema ya está resuelto.

En casi todos los casos, la capa que intentamos colocar va a tener que tener el position absolute, porque con relative no arreglamos totalmente el problema. Entonces volvemos a el problema inicial, que era situar la capa con position absolute en el lugar exacto, independientemente de la definición de pantalla.

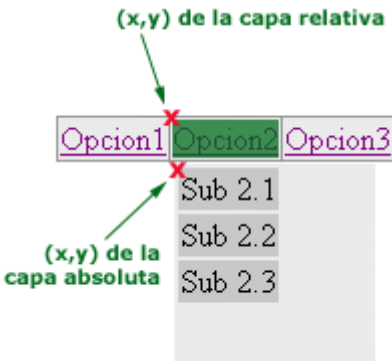
La solución final que propongo pasa por aplicar algún truquillo. De hecho, estuve hace unos días preguntándome sobre esa cuestión y al final encontré la solución, aunque no se me ocurrió a mi, sino que la extraje de www.cross-browser.com.

La idea es un poco compleja y para su puesta en marcha debemos realizar una serie de acciones que, sinceramente, considero excesivas para un problema inicialmente sencillo. Así pues, que no asuste lo que voy a soltar a continuación, que luego trataré de explicarlo un poco mejor.

Nuestro esquema de trabajo consistirá en una capa con posición relativa, que nos servirá de "ancla" y otra con la posición absoluta, donde colocaremos el contenido final a mostrar en la capa.

La capa relativa la colocaremos en el lugar aproximado donde queramos que aparezca la capa absoluta. La capa absoluta la posicionaremos, una vez cargada la página, en un lugar próximo a la capa relativa. Por supuesto, estas acciones las vamos a tener que realizar con Javascript, que es el lenguaje que nos permite actualizar las posiciones de las capas dinámicamente.

Detenidamente



Decíamos que habría que colocar una capa relativa cercana al lugar donde tiene que aparecer la capa con posición absoluta. Insisto en que las capas relativas se colocan en el lugar donde las metemos dentro del código HTML, por lo que será fácil colocar la capa relativa en el lugar exacto y que este lugar sea válido para cualquier definición.

La segunda capa, la que tiene el contenido final, la pondremos inicialmente en una posición cualquiera y escondida, de manera que no se vea que está mal colocada. Una vez terminada de cargar la página, podremos acceder a la posición de la capa relativa, extrayendo sus valores top y left y colocándolos en los correspondientes top y left de la capa con posición absoluta. Una vez marcada la posición de la capa absoluta podemos volverla visible.

A la vista de la imagen siguiente, la capa con posición relativa la hemos colocado en el enlace. En realidad habría tres capas con posición relativa para poder posicionar otras tantas capas con posición absoluta. La parte que vemos sombreada de verde corresponde al espacio que abarcaría la capa relativa.

Su posición sería la que está marcada por el aspa roja que aparece en su esquina superior izquierda. Dicha posición depende del lugar donde aparezcan los enlaces en la página.

Luego, con Javascript deberíamos asignar la posición de la capa absoluta de una manera parecida a esta.

left de la capa absoluta = left de la capa relativa
top de la capa absoluta = top de la capa relativa + altura de la capa relativa

Podemos sumarle algún píxel más a la posición de la capa, si es que queremos moverla un poco abajo y a la derecha, tal como hemos visto en la imagen.

No pretendo en este artículo, muy a mi pesar y por falta de espacio y tiempo, explicar cómo se hacen esas operaciones de Javascript. Advierto que si no se conoce nada de Javascript va a ser imposible ponerse con una tarea tan tediosa como el manejo de capas. Si por el contrario, ya hemos tenido contacto con Javascript y DHTML anteriormente, no debería ser un problema realizar esas acciones.

Referencias Javascript

En DesarrolloWeb tenemos un par de manuales de Javascript, que sería necesario estudiar para empezar a introducirse en el lenguaje.

- [Programación en Javascript I](#)
- [Programación en Javascript II](#)

En el [Taller de Javascript](#) tenemos algún artículo sobre tratamiento dinámico de capas.

En el momento en el que escribo estas líneas no tenemos un manual de DHTML y tratamiento de capas completo, por lo que sería recomendable estudiarlo en alguno de los enlaces recomendados de [nuestra sección de DHTML del buscador](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com



Qué es la programación orientada a objetos

La programación Orientada a objetos (POO) es una forma especial de programar, más cercana a como expresaríamos las cosas en la vida real que otros tipos de programación.

Con la POO tenemos que aprender a pensar las cosas de una manera distinta, para escribir nuestros programas en términos de objetos, propiedades, métodos y otras cosas que veremos rápidamente para aclarar conceptos y dar una pequeña base que permita soltarnos un poco con este tipo de programación.

Motivación

Durante años, los programadores se han dedicado a construir aplicaciones muy parecidas que resolvían una y otra vez los mismos problemas. Para conseguir que los esfuerzos de los programadores puedan ser utilizados por otras personas se creó la POO. Que es una serie de normas de realizar las cosas de manera que otras personas puedan utilizarlas y adelantar su trabajo, de manera que consigamos que el código se pueda reutilizar.

La POO no es difícil, pero es una manera especial de pensar, a veces subjetiva de quien la programa, de manera que la forma de hacer las cosas puede ser diferente según el programador. Aunque podamos hacer los programas de formas distintas, no todas ellas son correctas, lo difícil no es programar orientado a objetos sino programar bien. Programar bien es importante porque así nos podemos aprovechar de todas las ventajas de la POO.

Cómo se piensa en objetos

Pensar en términos de objetos es muy parecido a cómo lo haríamos en la vida real. Por ejemplo vamos a pensar en un coche para tratar de modelizarlo en un esquema de POO. Diríamos que el coche es el elemento principal que tiene una serie de características, como podrían ser el color, el modelo o la marca. Además tiene una

serie de funcionalidades asociadas, como pueden ser ponerse en marcha, parar o aparcarse.

Pues en un esquema POO el coche sería el objeto, las propiedades serían las características como el color o el modelo y los métodos serían las funcionalidades asociadas como ponerse en marcha o parar.

Por poner otro ejemplo vamos a ver cómo modelizaríamos en un esquema POO una fracción, es decir, esa estructura matemática que tiene un numerador y un denominador que divide al numerador, por ejemplo $3/2$.

La fracción será el objeto y tendrá dos propiedades, el numerador y el denominador. Luego podría tener varios métodos como simplificarse, sumarse con otra fracción o número, restarse con otra fracción, etc.

Estos objetos se podrán utilizar en los programas, por ejemplo en un programa de matemáticas harás uso de objetos fracción y en un programa que gestione un taller de coches utilizarás objetos coche. Los programas Orientados a objetos utilizan muchos objetos para realizar las acciones que se desean realizar y ellos mismos también son objetos. Es decir, el taller de coches será un objeto que utilizará objetos coche, herramienta, mecánico, recambios, etc.

Clases en POO

Las clases son declaraciones de objetos, también se podrían definir como abstracciones de objetos. Esto quiere decir que la definición de un objeto es la clase. Cuando programamos un objeto y definimos sus características y funcionalidades en realidad lo que estamos haciendo es programar una clase. En los ejemplos anteriores en realidad hablábamos de las clases coche o fracción porque sólo estuvimos definiendo, aunque por encima, sus formas.

Propiedades en clases

Las propiedades o atributos son las características de los objetos. Cuando definimos una propiedad normalmente especificamos su nombre y su tipo. Nos podemos hacer a la idea de que las propiedades son algo así como variables donde almacenamos datos relacionados con los objetos.

Métodos en las clases

Son las funcionalidades asociadas a los objetos. Cuando estamos programando las clases las llamamos métodos. Los métodos son como funciones que están asociadas a un objeto.

Objetos en POO

Los objetos son ejemplares de una clase cualquiera. Cuando creamos un ejemplar tenemos que especificar la clase a partir de la cual se creará. Esta acción de crear un objeto a partir de una clase se llama instanciar (que viene de una mala traducción de la palabra instace que en inglés significa ejemplar). Por ejemplo, un objeto de la clase fracción es por ejemplo $3/5$. El concepto o definición de fracción sería la clase, pero cuando ya estamos hablando de una fracción en concreto $4/7$, $8/1000$ o cualquier otra, la llamamos objeto.

Para crear un objeto se tiene que escribir una instrucción especial que puede ser distinta dependiendo el lenguaje de programación que se emplee, pero será algo parecido a esto.

```
miCoche = new Coche()
```

Con la palabra `new` especificamos que se tiene que crear una instancia de la clase que sigue a continuación. Dentro de los paréntesis podríamos colocar parámetros con los que inicializar el objeto de la clase coche.

Estados en objetos

Cuando tenemos un objeto sus propiedades toman valores. Por ejemplo, cuando tenemos un coche la propiedad `color` tomará un valor en concreto, como por ejemplo rojo o gris metalizado. El valor concreto de una propiedad de un objeto se llama estado.

Para acceder a un estado de un objeto para ver su valor o cambiarlo se utiliza el operador punto.

```
miCoche.color = rojo
```

El objeto es `miCoche`, luego colocamos el operador punto y por último el nombre e la propiedad a la que deseamos acceder. En este ejemplo estamos cambiando el valor del estado de la propiedad del objeto a rojo con una simple asignación.

Mensajes en objetos

Un mensaje en un objeto es la acción de efectuar una llamada a un método. Por ejemplo, cuando le decimos a un objeto coche que se ponga en marcha estamos pasándole el mensaje "ponte en marcha".

Para mandar mensajes a los objetos utilizamos el operador punto, seguido del método que deseamos imbobar.

```
miCoche.ponerseEnMarcha()
```

En este ejemplo pasamos el mensaje `ponerseEnMarcha()`. Hay que colocar paréntesis igual que cualquier llamada a una función, dentro irían los parámetros.

Otras cosas

Hay mucho todavía que conocer de la POO ya que sólo hemos hecho referencia a las cosas más básicas. También existen mecanismos como la herencia y el polimorfismo que son unas de las posibilidades más potentes de la POO.

La herencia sirve para crear objetos que incorporen propiedades y métodos de otros objetos. Así podremos construir unos objetos a partir de otros sin tener que reescribirlo todo.

El polimorfismo sirve para que no tengamos que preocuparnos sobre lo que estamos trabajando, y abstraernos para definir un código que sea compatible con objetos de varios tipos.

Son conceptos avanzados que cuesta explicar en las líneas de ese informe. No hay que olvidar que existen libros enteros dedicados a la POO y aquí solo pretendemos dar un repaso a algunas cosas para que os suenen cuando tengáis que ponerlos delante de ellas en los lenguajes de programación que debe conocer un desarrollador del web.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Tutorial de Visual Basic Script

Manual por: DesarrolloWeb.com

"Tu mejor ayuda para aprender a hacer webs"

Versión on-line:

<http://www.desarrolloweb.com/manuales/1>

Introducción a Visual Basic Script

El lenguaje para describir páginas, HTML, queda limitado a la hora de definir cualquier tipo de interactividad. Una vez que hemos explotado su potencia, estamos en necesidad de aprender algún lenguaje nuevo para hacer pequeños efectos o interactividades.

Scripts

Son los pequeños programitas que, incrustados en las páginas, nos permiten definir aquellos efectos o interactividades.

Visual Basic Script

En este manual nos vamos a ocupar de Visual Basic Script, un lenguaje compatible con Internet Explorer y otros sistemas Microsoft, por lo que en principio es una ventaja para programadores experimentados en estos sistemas.

Otros lenguajes

Existen dos tipos principales de lenguajes de scripting, y multitud de utilidades distintas, pero cabría destacar el lenguaje **Javascript** por ser parecido en utilidad a VBScript pero compatible con los dos navegadores más habituales.

Cómo poner scripts

Para poner un script en una página web utilizamos la etiqueta de HTML **<SCRIPT>**. Todo lo que pongamos entre esa etiqueta y la de cierre, **</SCRITP>**, tiene que ser código del lenguaje de scripting que estemos utilizando.

También debemos indicar el lenguaje con el que estamos programando. En nuestro caso pondremos:

```
<script language="VBScript" >  
---Aquí pondremos nuestros scripts---
```

`</script>`

Parece una tontería, pero fijaros que la palabra **language** en inglés se escribe con dos "G": **language**. Si os equivocáis en este punto, cosa bastante probable si escribís rápido y despistados, no funcionarán vuestros scripts pues el navegador pensará que están escritos en JavaScript.

Primer Script sencillo

Para terminar este capítulo vamos a ver un primer ejemplo de script en una página web. El objetivo de este script es mostrar la fecha de la última modificación del documento

```
<html>
<head>
  <title> La última modificación del documento</title>
</head>

<body>
<h1>Script de la última modificación de un documento</h1>
<script language="VBScript">
  document.write "Este documento fue actualizado por última vez en: "
  document.write document.lastmodified
</script>
</body>
</html>
```

La sentencia **document.write** es un procedimiento que escribe en la página web el texto que recibe por parámetro, el texto que esta después de la sentencia.

La variable **document.lastmodified** almacena la fecha y la hora de la última actualización.

Este script dará como resultado que el documento te informe de su última actualización, de una manera parecida a esta:



Este documento fue actualizado por última vez en: 03/07/2003 11:30:17

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Primeros pasos con el lenguaje

Los lenguajes de scripting tienen una serie de características comunes, estas suelen hacer la programación más fácil para personas inexpertas, pero a la larga pueden convertirse en una fuente de errores. Veamos cuáles son estas características, en concreto para VBScript.

Mayúsculas y minúsculas

En VBScript no importa si utilizamos mayúsculas o minúsculas a la hora de escribir nuestro código.

Variables

Las variables son espacios donde se almacenan los datos que utilizan los programas o scripts.

No se declaran: en VBScript las variables no se han de declarar, es decir, cuando necesitemos una variable, simplemente la utilizamos y ya está. Aún así, si deseamos declarar una variable utilizamos la palabra **DIM**

No hay tipos: las variables no están tipadas, esto quiere decir que podemos

guardar en ellas igualmente números que letras que otras cosas.

Salto de línea

Son importantes los saltos de línea. Expresan el final de una instrucción y el principio de la siguiente. No se pueden poner dos instrucciones en una misma línea.

Comentarios

En VBScript los comentarios se colocan con una comilla simple '. Esto sirve para que todo lo que se encuentre en esa línea después de la comilla simple sea ignorado por el explorador.

Ejemplo de todo esto

Vamos a ver a continuación un sencillo script que sirve de ejemplo para todo lo dicho anteriormente.

El ejemplo siguiente despliega unas ventanitas con mensajes (sentencia **msgbox**) siendo los mensajes el contenido de la variable **pepe**. Durante el ejemplo se cambia el valor de la variable y se vuelve a mostrar.

El ejemplo demuestra que no importan las mayúsculas y minúsculas y que es indiferente el tipo del contenido de la variable, texto o números.

```
<HTML>
<HEAD>
  <TITLE>Ejemplo2 Comentario, caja alert y variables</TITLE>
</HEAD>
<BODY>
  <script language=VBScript>
    'Esto es un comentario
    PEPE="HOLA"
    msgbox(pepe)
    pepe=3456
    'NO importan las mayusculas-minusculas
    msgbox(PEPE)
  </script>
</BODY>
</HTML>
```

[Pincha aquí para ver el ejemplo](#)

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Distintas formas de ejecutar scripts

Ahora que ya sabes cómo incluir scripts en tus páginas y unos cuantos fundamentos del lenguaje, vamos a ver los dos casos en los que Internet Explorer puede ejecutar tus scripts, de paso que le damos un primer vistazo a el concepto de evento.

Las formas de ejecución de VBScript son las siguientes:

- Scripts que se ejecutan mientras que el navegador abre la página.
- Scripts que se ejecutan como respuesta a la acción de un usuario.

El primero de los casos se utiliza cuando quieres hacer algo cuando el navegador carga la página. Por ejemplo, podrías mostrar un mensaje de bienvenida que aparezca cuando el usuario entra en tu página, o que el navegador te informe de la última actualización del documento (Tal como se vió en el capítulo 1).

El segundo caso es útil cuando deseas realizar acciones como respuesta a eventos del usuario.

Los **eventos** son acciones que ocurren cuando un usuario hace alguna cosa sobre la página web, es decir, un evento podría ser que el usuario escriba algo en una caja de texto, o que se coloque con el ratón encima de un enlace, y así un montón de cosas. Casi cualquier cosa que puede realizar el usuario dentro de la página tiene un evento relacionado.

Utilizando los eventos podemos preparar algún efecto que sea solo visible cuando el usuario realice alguna acción dentro de la página web.

Ejemplo de todo esto

Veamos ahora un ejemplo para acabar de comprender las dos formas de ejecución de los scripts

Vamos hacer que el navegador nos diga su número de versión y otros datos en un cuadro de diálogo. Lo vamos a hacer de las dos maneras, según se carga la página y cuando el usuario aprete un botón.

Ejemplo de ejecución al cargar la página

Comencemos por la ejecución de scripts cuando el usuario carga la página. Esta es la forma más sencilla, y realmente ya conoces varios ejemplos de esto que viste en los anteriores capítulos.

```
<html>
<head>
  <title>Escript de ejecución directa</title>
</head>
<body>
  Según se carga la página vamos a ver
  la versión del navegado en una caja de diálogo.
  <script language=vbscript>
    msgbox(navigator.appVersion)
  </script>
</body>
</html>
```

Este ejemplo no tiene ningún misterio, pues es muy parecido a los ejemplos que hemos realizado. la única novedad es la variable **navigator.appVersion**. Esta almacena lo que queremos que se vea en la caja de diálogo: la versión del navegador. [Pincha aquí para ver el ejemplo](#)

Ejemplo de ejecución como respuesta a la acción del usuario

Ahora veamos lo que hay que hacer cuando deseamos que esta caja de diálogo no aparezca hasta que el usuario pulse en un botón.

```
<html>
<head>
  <title>Script de ejecución como respuesta a un evento</title>
</head>
<body>
  Pulse el botón para ver la versión del navegador
  <input type=button value=Pulsame onclick="msgbox(navigator.appVersion)"
  language="vbscript">
</body>
</html>
```

Este ejemplo tiene cosas nuevas que habría que destacar:

1. Se crea un botón con la etiqueta <INPUT>

2. Se le añade el atributo **onclick**. Este sirve para indicar (en lenguaje de script) las acciones que queremos realizar como respuesta al evento "click sobre el botón".
3. Se le añade el atributo **language** para especificar el lenguaje en el que está escrito el código script asociado al evento.

Ahora tenemos un botón que, cuando se pulse, ejecutará el código que despliega una caja de diálogo con la versión del navegador.

[Pincha aquí para ver el ejemplo](#)

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Declarar variables

Antes de pasar a temas más interesantes queda explicar la forma de declarar variables en VBScript. Hace poco se dijo que no era necesario declarar estas variables, pero puede ser una buena costumbre hacerlo y nos puede evitar errores.

Una variable se declara utilizando la palabra **DIM**, veamos cómo:

```
<script language="vbscript">  
  dim mi_nueva_variable  
  'Ahora ya existe la variable  
  'Seguidamente voy a hacer uso de ella  
  mi_nueva_variable = "Valor de la variable"  
</script>
```

Como se ha de recordar no importa que tipo de información va a contener la variable, siempre se declaran igual.

Option explicit

Se puede utilizar la clausula **Option explicit** para forzar la declaración de variables en nuestros scripts. Si deseas evitar la posible fuente de errores que supone la libertad de no declarar las variables puedes utilizar esta clausula y hará que tus scripts respondan con mensajes de error si utilizas una variable que no has declarado previamente. Veamos su uso con un ejemplo:

```
<HTML>  
<HEAD>  
  <TITLE>Option explicit</TITLE>  
</HEAD>  
<BODY>  
  <script language=vbscript>  
    option explicit  
  
    dim Pepe  
    pepe = 3  
    tomas = 87  
  </script>  
</BODY>  
</HTML>
```

Este script responderá con un mensaje de error cuando se ejecute, pues la variable tomas no se ha declarado antes de su uso.

Informe de **Miguel Angel Alvarez**

Tipos de datos

Visual Basic Script posee varios tipos de datos pero en la práctica sólo posee un tipo de variable, que va cambiando de un estado a otro según la información que introducimos dentro. Este tipo *principal* de datos es el tipo **Variant**, en él podemos introducir varios *subtipos* de datos con total libertad.

Para cambiar el subtipo de un variant, sólo tenemos que introducir un dato en la variable. La variable variant cambia automáticamente de un subtipo a otro, sin que tengamos que hacer ninguna operación adicional. Los distintos subtipos de datos que tenemos son los siguientes:

Booleano	Es un tipo de datos que contiene un si o un no. se corresponden: TRUE equivale a (-1) FALSE equivale a (0)
Byte	Numérico, entero sin signo hasta 65.000
Currency	Tipo de moneda , se utiliza para manipular de manera exacta valores monetarios, y en general cualquier cálculo que requiera una precisión de hasta 15 dígitos decimales
Fecha	Es un tipo de 64 bits de tamaño que almacena fechas. Se utiliza el formato americano: mes, día, hora.
Double	Coma flotante con doble precisión (64 bits)
Entero	Número entero, con signo. Desde -32.768 hasta 32.767
Entero largo	Este tipo es un valor entero con signo de doble precisión. Como los nuevos ordenadores trabajan con palabras de 32 bits, y no menos, se recomienda usar este tipo antes de el tipo entero normal.
Objeto	El subtipo de objeto es una referencia de puntero de 32 bits a una instancia de de objeto de automatización OLE. Los controles Active-X y java. Utilizan esta sintaxis: Set miobjeto = new oleObjeto
Single	Coma flotante de precisión simple
Cadena	Conjunto continuo de valores de caracteres, de longitud variable.

Como saber de qué subtipo es una variable

Para averiguar el subtipo de una variable podemos utilizar la función VarType, de esta manera:

```
v1 = 3
```

```
document.write VarType (v1)
```

En este script declaramos una variable y le metemos un número y a continuación imprimimos en la página el valor que devuelve la función VarType.

Al [ejecutar este script](#) podremos ver un "2" escrito en la página.

Según el tipo de datos que halla en la variable, VarType devolverá un valor distinto, como indica esta tabla:

Constant	Value	Description
vbEmpty	0	Empty (uninitialized)
vbNull	1	Null (no valid data)
vbInteger	2	Integer
vbLong	3	Long integer
vbSingle	4	Single-precision floating-point number
vbDouble	5	Double-precision floating-point number
vbCurrency	6	Currency
vbDate	7	Date
vbString	8	String
vbObject	9	Automation object_
vbError	10	Error
vbBoolean	11	Boolean
vbVariant	12	Variant (used only with arrays of Variants)
vbDataObject	13	A data-access object
vbByte	17	Byte
vbArray	8192	Array

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Operadores I - Aritméticos

Visual Basic Script, como cualquier lenguaje de programación, tiene un conjunto de operadores, divididos en varias secciones:

Operadores Aritméticos

Que soportan las operaciones matemáticas más sencillas.

+	Suma
-	Resta
*	Multiplicacion
/	División en coma flotante. Es la división normal. Devuelve un numero real si es el resultado
\	División de enteros

	Devuelve un numero entero, resultado de la división.
^	Potencia
Mod	Resto de la división

Veamos a continuación un ejemplo de script que realiza acciones con estos operadores:

```

dim v1

dim v2

v1 = 34

v2 = 43

suma = v1 + v2

resta = v1 - v2

potencia = v1 ^ v2

divisionEnteros = v1 \ v2

msgbox (divisionEnteros)

DivisionReal = v1 /v2

msgbox (divisionReal)

```

La función msgbox sirve para mostrar un valor en una ventanita de alerta típica de Windows. Podemos [ver este script en funcionamiento](#).

Informe de **Miguel Angel Alvarez**
 Director desarrolloweb.com
 Mail: eugim@desarrolloweb.com

Operadores II - Comparación

Para realizar comparaciones, Visual Basic Script posee los siguientes operadores:

=	<>	Igual y distinto
>	<	Mayor que y menor que
>=	<=	Mayor o igual que y menor o igual que

Veamos a continuación un ejemplo de script que realiza operaciones de comparación, aunque antes que verlo deberíamos puntualizar que los operadores de comparación se suelen utilizar dentro de una estructura condicional, que evalúa una expresión con estos comparadores y realiza acciones dependiendo del resultado de esas comparaciones. Por este motivo hemos incluido en el script la estructura condicional IF que veremos con profundidad más adelante.

```

precio = 20000

```

```

dineroActual = 3500

if (dineroActual = precio) then
    msgbox ("Lo tienes justo")
end if

if (dineroActual < precio) then
    msgbox ("te falta dinero")
end if

```

Podemos [ver este script en funcionamiento](#).

Informe de **Miguel Angel Alvarez**
 Director desarrolloweb.com
 Mail: eugim@desarrolloweb.com

Operadores III lógicos y cadenas

Operadores lógicos

Los operadores lógicos se utilizan sobre expresiones booleanas y nos devuelven un valor booleano (verdadero o falso) resultado de esa operación. Un matiz sería que no es necesario que los operandos relacionados en el cálculo sean variables booleanas, pudiendo ser de cualquier tipo.

Como operadores lógicos en Visual Basic Script tenemos:

AND	Y lógico
OR	O lógico
Xor	Xor
Not	NO lógico

Operadores de cadenas

Como operador de cadenas de caracteres en Visual Basic Script tenemos un único ejemplo: la concatenación. El operador para concatenar cadenas es el **&**. Veamos un ejemplo de utilización de este operador:

```

cadena1 = "Hola "
cadena2 = "Pepe"

concatenar = cadena1 & cadena2

msgbox (concatenar)

```

Podemos [ver este script en funcionamiento](#).

Informe de **Miguel Angel Alvarez**
 Director desarrolloweb.com

Estructuras de control

Las estructuras de control nos permiten realizar acciones típicas en nuestros scripts como lo pueden ser los bucles o la toma de decisiones.

VBScript tiene las estructuras de control típicas de los lenguajes de programación. Vamos a ver la sintaxis y la manera de trabajar de estas estructuras detenidamente:

Condicionales

- **IF**, condicional que decide entre si/no
- **CASE**, otro condicional con varias posibilidades

Bucles

- **FOR**, repetición un determinado número de veces
- **FOR EACH**, repetición para un conjunto de elementos
- **WHILE...WEND**, repetición mientras ocurra alguna cosa
- **DO...LOOP**, repetición un determinado número de veces

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Estructura IF

La estructura de control IF permite decidir entre dos opciones resultantes de la evaluación de una sentencia. Si la evaluación es positiva hace una cosa, también podemos especificar acciones para realizar en caso de que la evaluación sea negativa. Veamos cómo funciona en VBScript.

```
IF (expresion) then
    Sentencias
    ....
END IF
```

Vemos que en primer lugar tenemos la sentencia IF, luego una expresión, que puede o no ir entre paréntesis, y más tarde la palabra THEN. Vemos que luego hay un salto de línea antes de colocar las sentencias asociadas a la evaluación positiva de la sentencia. En VBScript las líneas si que importan.

Después de poner las sentencias del asociadas a la evaluación positiva colocamos un END IF, para acabar la estructura del IF.

Enunciado ELSE

Opcionalmente se puede colocar una serie de sentencias asociadas a la evaluación negativa de la expresión. Estas sentencias se deben colocar después de la orden ELSE y antes del END IF.


```

IF (expresion) then
    Sentencias
    ....
ELSE
    Sentencias
    ....
END IF

```

Enunciado ELSEIF

En Visual Basic Script existe la posibilidad de utilizar un enunciado especial en el lugar donde utilizaríamos un ELSE. Sirve para encadenar sentencias IF de modo que en un resultado negativo de un IF se pueda evaluar otra expresión, que tendría a su vez otros enunciados THEN y probablemente ELSE u otro ELSEIF. Se vería en un ejemplo con más facilidad:

```

IF (expresion1) then
    Sentencias1
    ....
ELSEIF (expresion2) then
    Sentencias2
    ....
ELSEIF (expresion3) then
    Sentencias3
    ....
ELSE
    Sentencias4
    ....
END IF

```

Se evalúa la primera expresión, en caso positivo se ejecutan las sentencias1, en caso negativo se evalúa la expresión 2. Si la expresión 2 es positiva se ejecutan las sentencias 2 en caso negativo evaluamos la expresión 3 con el siguiente ELSEIF. Todo acaba en un ELSE en este ejemplo, pero el ELSE final no es obligatorio.

Informe de **Miguel Angel Alvarez**
 Director desarrolloweb.com
 Mail: eugim@desarrolloweb.com

Estructura CASE

Con la estructura de control CASE podemos evaluar una variable y realizar acciones dependiendo del valor de esta. La diferencia con el IF consiste en que el número de posibilidades de la evaluación de esta variable no tiene por que ser si o no, pudiendo hacer cosas para un número indeterminado de valores.

La sintaxis es la siguiente:

```

SELECT CASE (variable)
    CASE (valor1):
        (acción para caso valor1)
    CASE (valor2):
        (acción para caso valor2)
    CASE (valor3):
        (acción para caso valor3)
    CASE ELSE:
        (acción en caso de que no se cumpla ningun anterior caso)
END SELECT

```

Funciona así, primero se evalúa la variable, si esa variable tiene como valor el valor1 realizamos las acciones asociadas al valor1. Si tiene el valor2, ejecutamos las acciones relacionadas con este valor3. Así con cuantos valores deseemos. Por último tenemos un ELSE para realizar acciones en caso de que no hubiesen sido ninguno de los valores anteriores. Este ELSE es opcional.

Veamos con un ejemplo esta sentencia muy sencillito. Lo primero que hace es solicitar un número y luego informa del día de la semana con el que corresponde. Si el número no es del uno al siete informa de ello también.

```
dim dia
dia = inputbox ("dime un dia de la semana")
SELECT CASE dia
CASE 1:
  msgbox("El dia es LUNES")
CASE 2:
  msgbox("El dia es MARTES")
CASE 3:
  msgbox("El dia es MIERCOLES")
CASE 4:
  msgbox("El dia es JUEVES")
CASE 5:
  msgbox("El dia es VIERNES")
CASE 6:
  msgbox("El dia es SABADO")
CASE 7:
  msgbox("El dia es DOMINGO")
CASE ELSE:
  msgbox("Tiene que ser un dia de la semana en número, del 1 al 7")
END SELECT
```

Este script se puede [ver en funcionamiento](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Bucle FOR

La sentencia FOR se utiliza para los bucles, cuando sabemos el número de veces que debemos ejecutar el bucle. Veamos su sencilla sintaxis:

```
FOR (inicializacion) TO (termino del bucle) STEP (paso)
  sentencias
  .....
NEXT
```

La sentencia realiza una repetición desde la *inicialización* hasta el *término del bucle*. Para llevar la cuenta se utiliza una variable, ya veremos en el ejemplo cómo se utiliza esta variable. Con cada ejecución del bucle se ejecutan unas sentencias. NEXT sirve para delimitar el final del bucle, cuando se encuentra con el NEXT se vuelve otra vez al principio del FOR, así hasta realizar el número de ejecuciones determinado.

Existe un valor que sirve para indicar los grandes que se desean realizar los saltos entre ejecución y ejecución, es el valor STEP. Un STEP 2 determinaría que entre ejecución y ejecución la variable se ha de incrementar en 2 unidades. En el caso de

no indicar nada se realizan pasos de 1 en 1. También podemos realizar pasos en valores negativos.

Un ejemplo de estos datos sería el siguiente:

```
for i=0 to 6 step 2
    msgbox(i)
next
```

Este ejemplo presentaría un mensaje con un numerito de la variable i, utilizada para llevar la cuenta de las ejecuciones del bucle. Se puede ver un ejemplo de [este script funcionando](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Bucle FOR EACH

La estructura de control FOR EACH sirve para moverse por los elementos de una estructura de datos, como podría ser un vector, y realizar acciones para cada una de los elementos.

Veamos con un ejemplo esta estructura de control: En el ejemplo primero creamos un vector y rellenamos con números cada una de sus casillas, con un bucle FOR normalito. Más tarde utilizamos el bucle FOR EACH para acceder a cada una de las posiciones de este vector de números y escribir en la página cada una de estos números.

```
dim tor(20)
for i=0 to 20
    tor(i) = i
next
for each i in tor
    document.write (tor(i))
next
```

Fijemonos en el segundo bucle, se indica que para cada i (i es el índice con el que podemos movernos en el bucle y en la estructura) dentro de tor (que es la estructura, en este caso un vector) haga un document.write(tor(i)). Con tor(i) accedemos a la casilla actual y document.write() sirve para escribir algo en la página web. Conbinadas lo que se escribe es lo que hay en la posición actual del vector. Se puede ver [este script funcionando](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Bucle WHILE WEND

El bucle WHILE...WEND sirve para realizar un tipo de bucle mu utilizado en

programación que es el bucle Mientras, que se ejecuta mientras que se cumpla una condición. A diferencia del bucle FOR, éste se utiliza cuando no conocemos el número de iteraciones que tenemos que realizar.

El bucle funciona de la siguiente manera. Cuando se va a ejecutar, evalúa una expresión y comprueba que esta da resultados positivos. Si es así, ejecuta el cuerpo del bucle (las sentencias que siguen hasta el WEND), en caso contrario se sale.

Podemos ver la sintaxis a continuación.

```
WHILE (condicion)
    sentencias
    ....
WEND
```

Ahora vamos a ver un ejemplito sobre este bucle, que realiza una cuenta número a número hasta llegar al 13. En cada iteración del bucle muestra en una ventanita el número actual y ofrece la posibilidad de cambiarlo, ya que la ventanita es una ventana Input, que ofrece la oportunidad de cambiar el valor y devuelve ese valor, cambiado o no. Como decíamos, si dejamos el ejemplo sin tocar nada, cuenta hasta 13, pero si introducimos un número en el inputbox continúa la cuenta por el número introducido. Si el número introducido es mayor que 13 también se sale del bucle.

```
option explicit
dim a
a = 0
WHILE (a < 13)
    a = a + 1
    a = inputbox("Dame un valor entero, please", "Petición de número",a,200,100)
WEND
```

Podemos [ver este ejemplo en funcionamiento.](#)

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Bucle DO LOOP

El bucle DO...LOOP es muy versátil. Con él se pueden crear gran variedad de bucles distintos, bucles que comprueben una condición antes de ejecutar el bucle una vez, después de la primera ejecución y con combinaciones con mientras (WHILE) que se cumple una condición o hasta (UNTIL) que esa condición se cumpla. La sintaxis de esta estructura es la siguiente:

```
DO [WHILE | UNTIL (condicion)]
    Sentencias
    .....
LOOP [WHILE | UNTIL (condicion)]
```

Vamos a tratar de explicar esta sentencia de manera pausada para que sea más fácil de entender. Lo que siempre tendremos en estos bucles es el DO y el LOOP, entre estos dos colocaremos las sentencias que queremos ejecutar en cada iteración del bucle. Los bucles tienen que evaluar entre cada iteración si se siguen ejecutando o no, para ello evalúan una condición. Lo versátil de este bucle es que la condición se puede expresar de muchas maneras distintas.

Condición expresada al lado del DO: en este caso la condición se evalúa antes de empezar a ejecutarse el bucle.

Condición expresada al lado del LOOP: en este caso la condición se evalúa después de ejecutarse el bucle. Tiene como diferencia principal frente al otro método que en este caso el bucle se ejecutará por lo menos una vez.

Además de poder expresar la condición en estos dos sitios también se puede construir la condición con un enunciado mientras (WHILE) o un enunciado hasta (UNTIL). Las diferencias semánticas de estas dos posibilidades se trasladan también a su manera de funcionar.

Vamos a ver un par de ejemplos de este bucle para comprender su funcionamiento. El ejemplo pide constantemente el nombre del autor de la página y no para hasta que el nombre sea "migue". También tiene el usuario la posibilidad de escribir "out", en ese caso, comprobado con un enunciado IF, se sale del bucle rompiéndolo con la sentencia EXIT DO, utilizada para romper bucles.

```
Dim entrada
entrada = ""
DO WHILE (entrada <> "migue")
  entrada = inputbox ("Dime el nombre del autor", "seguridad", "migue", 2, 3)
  if (entrada = "out") then
    msgbox "salgo por la puerta de atrás"
    exit do
  end if
LOOP
```

Podemos [ver este ejemplo en funcionamiento](#).

El siguiente ejemplo realiza una cuenta y entre cuenta y cuenta se muestra el valor de la cuenta actual en una ventanita donde sale un botón de Reintentar y otro de Cancelar. Si se pulsa reintentar se sigue ejecutando el bucle y si se pulsa Cancelar se sale por la puerta de atrás, de manera similar a como se salía en el ejemplo anterior, con EXIT DO.

```
option explicit
dim cont
dim respuesta
cont = 0
DO
  cont = cont + 1
  respuesta = msgbox (cont, 69, "Variable del bucle, con valor 6 se sale")
  if (respuesta = 2) then
    msgbox "Cuenta Cancelada", 16, "Cancelaste!"
    exit do
  end if
LOOP UNTIL (cont = 6)
```

El ejemplo es un poco raro, pero servirá para comprender estos bucles. Se puede [ver en funcionamiento](#) para entender mejor la manera de ejecutarse que tiene este bucle.

Arrays en Visual Basic Script

Los Arrays o matrices son unas estructuras de datos muy utilizadas en cualquier lenguaje. Se tratan de variables, pero que están preparadas para guardar una cantidad mayor de elementos. Es como una variable que tiene varios compartimentos para guardar la información y a cada uno de esos compartimentos hay que acceder con un índice.

Antes de utilizar un array debemos declararlo de manera obligatoria, para ello utilizamos la palabra clave DIM, de este modo.

```
dim miArray(20)
```

Después de la palabra DIM debemos indicar el nombre del array y a continuación, entre paréntesis, se coloca el número de posición máxima del array, en este caso 20.

Los arrays en ASP comienzan desde la posición 0, es decir, el primer elemento de un array está en la posición 0. Por tanto, si el array ha sido definido con 20 casillas, como en el ejemplo, tendrá 21 elementos, primera posición será la 0 y la última posición sería la 20.

Para asignar un valor a un array se realiza igual que una variable, pero accediendo con el índice de la posición que queremos escribir.

```
miArray(0) = 234
```

Para utilizar el contenido de un array debemos hacerlo indicando el índice al que se desea acceder. Por ejemplo, si quisiésemos imprimir en la página la primera posición de nuestro Array lo haríamos de esta manera.

```
document.write(miArray(0))
```

Ahora vamos a ver un ejemplo sobre cómo utilizar los arrays, donde vamos a realizar dos recorridos, uno para escribir en él y el otro para leer la información y escribirla en la página.

```
dim matriz (10)
for i=0 to 10
    matriz(i)=100 * i
next

for i=0 to 10
    document.writeln("Posicion " & i & ": " & matriz(i) & "<br>")
next
```

Este ejemplo escribiría en la página las posiciones del array, que contienen variables numéricas que corresponden de multiplicar su índice por 100.

Si se desea, se puede [ver el efecto resultante](#) en una página web.

Arrays multidimensionales

Se pueden construir matrices multidimensionales, es decir, que nos permitan crear matrices de varias coordenadas. Para trabajar con ellos se utiliza una coma que

separa los dos índices. Por ejemplo podemos definir una matriz de 8x8 de esta manera.

```
dim miArray2Dimensiones (7,7)
```

Como el array es de 8 casillas, utilizamos un 7 y sus posiciones serán las 8 que van desde el 0 al 7. Para escribir y leer del Array podemos utilizar la coma de manera similar a como se declara. Por ejemplo, para meter datos en la posición 0,2 haríamos lo siguiente:

```
miArray2Dimensiones (0,2) = "texto posicion 0,2"
```

Redimensionar arrays

Podemos declarar también arrays que cambien el número de casillas que tienen según se necesite en tiempo de ejecución. Este tipo de arrays redimensionables se llama array dinámico. Para crear este tipo de arrays podemos utilizar la sentencia `dim` (como creábamos los anteriores) o la sentencia `redim`, con la particularidad que no le colocamos ningún valor entre paréntesis donde antes indicábamos el número de casillas del array.

```
dim mi_array()  
redim mi_otro_array()
```

Cuando usamos arrays dinámicos podemos utilizar la sentencia `redim` para indicar el número de dimensiones y la cantidad de casillas de cada dimensión.

Con esta sentencia estamos indicando que `mi_array` debe tener el tamaño 10. Casillas desde la 0 hasta la 10.

```
redim mi_array(10)
```

Si indicamos la clave "preserve" estamos asegurándonos que el contenido de las casillas que había previamente en el array se mantiene.

```
redim preserve mi_array(20)
```

Por último, si en cualquier momento reducimos el número de casillas perderemos lo que pudiera haber guardado en las casillas que se han eliminado.

En el siguiente ejemplo creamos un array dinámico y lo redimensionamos inicialmente a tamaño 3. Lo rellenamos y mostramos sus distintos valores. Posteriormente lo redimensionamos otra vez para que llegue hasta la posición 7, guardando los valores antiguos. Para acabar rellenamos las casillas que hemos creado nuevas y mostramos todos los valores del array.

```
dim frutas()  
redim frutas(3)  
  
frutas(0) = "Pera"  
frutas(1) = "Uva"  
frutas(2) = "Manzana"  
frutas(3) = "Melón"  
  
for each fruta in frutas  
    document.write fruta & "<br>"  
next
```

```
redim preserve frutas(7)

frutas(4) = "Sandía"
frutas(5) = "Naranja"
frutas(6) = "Plátano"
frutas(7) = "Mandarina"

for each fruta in frutas
    document.write fruta & "<br>"
next
```

El ejemplo se puede [ver en funcionamiento en una página nueva](#).

Obtener el número de casillas de un array

Uno de los datos típicos que necesitamos extraer de un array es su número de posiciones, útil por ejemplo para hacer un recorrido a un array, desde la primera hasta la última casilla. Para ello utilizamos la función **uBound()** de VBScript.

uBound() recibe el array del que queremos obtener su número de posiciones y devuelve la posición más alta del array. Por ejemplo.

```
dim ciudades(5)
document.write ubound (ciudades)
```

Escribiría en la página el número de la casilla más alta del array ciudades, en este caso 5.

Además, por si algún día la necesitamos, también tenemos a nuestra disposición la función **lBound()**, que devuelve el número de la posición con índice menor del array.

```
document.write lbound (ciudades)
```

La última línea sobre nuestro array de ciudades definido anteriormente escribiría un 0 en la página web, puesto que el array comienza en la posición cero.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Procedimientos y funciones

Los procedimientos o funciones son muy interesantes y útiles en la programación. Nos sirven para realizar una tarea concreta que probablemente se vaya a ejecutar varias veces a lo largo de la vida de la página. Esta tarea se especifica en un bloque de código de manera independiente y cuando se desean realizar las acciones del procedimiento se llama al procedimiento o función. Una vez realizadas las acciones pertinentes se devuelve el flujo del programa al lugar desde donde se invocó ese procedimiento o función.

Lo primero que debemos hacer al crear un procedimiento es pensar las cosas que se desean hacer dentro de la función, la información que necesitaremos (y que tendremos que recibir como parámetros) y la información que devolverá. Con estas

ideas claras se pueden construir los procedimientos y funciones sin mucha dificultad, siguiendo estas estructuras.

Para un procedimiento

```
Sub nombre (parametro1, parametro2...)  
  ... Código del procedimiento  
end Sub
```

Para una función

```
Function nombre (parametro1, parametro2...)  
  ... Código de la función  
end Function
```

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Procedimientos. SUB

Decíamos que un procedimiento era una subrutina que se llamaba y realizaba acciones, pero que no devolvía ningún valor y por lo tanto, no era posible utilizarla dentro de una expresión.

Veamos algún ejemplo de procedimiento. Es una subrutina que escribe en la barra de estado un mensaje. No es muy complicada, pero tal como la presentamos aquí no se debería hacer, puesto que utilizamos un bucle vacío para que el navegador esté un poco más lento y el texto salga poco a poco. En lugar de ese bucle deberíamos utilizar una función llamada `setTimeout`, pero no deseamos introducirla ahora.

```
sub muestraAbajo(texto)  
  dim i  
  for i=0 to len(texto)  
    dim actual  
    actual = left(texto,i)  
    window.status = actual  
    dim j  
    'bucle para ralentizar al navegador debería utilizarse la función setTimeout  
    for j=0 to 20000  
      j = j  
    next  
  next  
next  
end sub
```

Este ejemplo utiliza además varias funciones de cadenas de caracteres, esperamos que no represente mucho problema para entenderlo. Básicamente es un bucle que va recorriendo toda la cadena de caracteres que recibe por parámetro. A medida que se realiza el bucle se va creando una subcadena de caracteres de la parte izquierda de la cadena original, que cada vez es más larga. Luego se imprime esa cadena en la barra de estado del navegador. Entre ejecución y ejecución del bucle

se realiza un retardo, en el segundo bucle for que se debería realizarse con un setTimeout.

Podemos ver a continuación cómo se colocaría un botón en la página que llamase a este procedimiento.

```
<HTML>
<HEAD>
<TITLE>Procedimientos en VBS</TITLE>
<script language=vbscript>
option explicit
sub muestraAbajo(texto)
dim i
for i=0 to len(texto)
dim actual
actual = left(texto,i)
window.status = actual
dim j
for j=0 to 20000
j = j
next
next
end sub
</script>
</HEAD>
<BODY>
<h1>Procedimientos en VBS</h1>

<P>
<form>
<input type="button" name=b value=ponerAbajo!
onclick="muestraAbajo('Saludos de Miguel')" language=vbscript>
</form>
</P>
</BODY>
</HTML>
```

Se puede [ver el ejemplo en una página aparte](#).

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com

Funciones. Function

Ya vimos lo que consistía una función, que no es más que un trozo de código que opera para devolver un valor. Ahora vamos a ver con detenimiento un ejemplo de su uso.

Vamos a definir una función que realice un cálculo matemático y devuelva el resultado del mismo. Los operandos los vamos a extraer de un formulario. El ejemplo puede ser ahora mismo un poco complejo, por tratar con formularios -que no hemos visto todavía-, pero podemos ver el código de la función y hacernos una idea exacta de su uso, que al fin y al cabo es lo que nos importa.

El código de la función será el siguiente:

```
function operar (operador,op1,op2)
```

```

select case operador
  case "+":
    operar = op1 + op2
  case "-":
    operar = op1 - op2
  case "*":
    operar = op1 * op2
  case else:
    operar = op1 / op2
end select
end function

```

Vemos que la función recibe tres parámetros, el primero es un operador, que no es más que un texto con el signo de la operación a realizar. Los dos siguientes parámetros son los operandos que hay que tratar.

La función realiza una operación matemática dependiendo de del operador y devuelve en cada caso el resultado conveniente. Fijémonos que para devolver un valor se debe realizar una asignación del nombre de la función al valor que se desea devolver.

No creemos que revista ninguna complicación. Vamos a ver ahora el código que podríamos utilizar para hacer la llamada a la función.

```

miOperador="+"
miOperando1=221
miOperando2=32
resultado = operar(miOperador,miOperando1,miOperando2)

```

Al final de todas estas sentencias la variable resultado tendrá como valor 253.

Veamos el ejemplo completo, que consistía en una calculadora hecha con un formulario, que usa esta función para obtener los resultados.

```

<HTML>
<HEAD>
<link rel=stylesheet type=text/css href=estiloglobal.css>
<TITLE>Funciones en VBS</TITLE>
</HEAD>
<h1>Funciones en VBS</h1>
<script language=vbscript>
function operar (operador,op1,op2)
select case operador
  case "+":
    operar = op1 + op2
  case "-":
    operar = op1 - op2
  case "*":
    operar = op1 * op2
  case else:
    operar = op1 / op2
end select
end function

sub opera ()
dim res
operador = document.forms(0).operacion.value
operando1 = cint(document.forms(0).op1.value)
operando2 = cint(document.forms(0).op2.value)
res = operar (operador,operando1,operando2)
document.forms(0).result.value = res

```

```

end sub
</script>
<BODY>
<form>
Operando 1
<input name=op1 >
<br>
Operando 2
<input name=op2 >
<br>
operacion:
<select name=operacion>
<option value="+" selected>+
<option value="-">-
<option value="*">*
<option value="/">/
</select>
<input type=button name=b value="realizar operacion"
onclick=opera language=vbscript>
<br>
Resultado:
<input name=result >
</BODY>
</HTML>

```

Hemos tenido que utilizar un procedimiento de apoyo para hacer el ejercicio, ya que, en caso de no utilizarlo, haría un poco más compleja a la función. Podremos entenderlo todo ya que no reviste mucha complicación y los [procedimientos los pudimos ver en el capítulo anterior](#). Para tener más claro todavía cómo trabaja esta página de ejemplo podemos [ver el ejemplo funcionando](#).

Tenemos un formulario donde podemos ver campos para los operadores, una caja de selección para el operando y un último campo para el resultado. Es interesante también el botón de realizar operación, que es el que lo pone todo en marcha gracias a su manejador de evento onclick, que quiere decir que cuando se pulse sobre el botón se realice una acción. En este caso es una llamada al procedimiento opera.

En el procedimiento opera podemos ver varias sentencias para extraer la información del formulario y también la llamada a la función que realiza los cálculos. Por último, se introduce en el campo resultado lo que devolvió la función como resultado de realizar las operaciones.

Podemos [ver el ejemplo en funcionamiento](#).

Informe de **Miguel Angel Alvarez**
 Director desarrolloweb.com
 Mail: eugim@desarrolloweb.com

Más sobre procedimientos y funciones

Ahora vamos a ver algunas cosas más sobre subrutinas que nos han quedado en el tintero. Un poco en cajón de sastre.

Llamadas a subrutinas

En Visual Basic Script las funciones se utilizan como partes de expresiones y los

procedimientos como si fuera una sentencia independiente.

La llamada a una función, si se utiliza como parte de una expresión se debe llamar utilizando paréntesis.

```
miResultado = suma(1,2)
```

Si no se utiliza como parte de una expresión, no tienen por que utilizarse los paréntesis, pero el resultado de la función (lo que devuelve) se perderá.

```
suma 1,2
```

Call

Es una llamada a una subrutina, utilizada para transferir el flujo de la aplicación hacia una subrutina. Es necesario utilizar paréntesis cuando se utiliza. Además, si se utiliza con una función se perderá el resultado que devuelva.

```
call suma(1,2)
```

Salida de una subrutina

Podemos salirnos de un procedimiento o función en cualquier momento, independientemente de que la función haya terminado o no. El enunciado para escaparse de una función es EXIT, que se puede utilizar en cualquier lugar del procedimiento o función. La palabra exit debe ir acompañada del tipo de subrutina de la que se desea salir, así pues se deberá utilizar o bien **exit function** o **exit sub**.

Informe de **Miguel Angel Alvarez**
Director desarrolloweb.com
Mail: eugim@desarrolloweb.com
