



Tutorial de CSS



b2evolution

Volumen 3, nº 3

Marzo, 2007

CONTENIDOS

Hola Cibernautas, en ésta oportunidad aquí en *ProXión* tenemos algunas novedades que te resultarán interesantes, entre ellas, como dice allí arriba: *Tutorial de CSS*, cortesía de la página psicobyte.com.

Adicionalmente también podrás encontrar:

WEB

Gestor de Contenidos: b2evolution.

PHOTOSHOP

Pequeño tutorial para crear Bordes como PixelTool.

SOFTWARE

PC Inspector File Recovery.

Esto es...

PROXIÓN 3

SEGURIDAD: Antivirus

ARCAVIR: www.arcabit.com

ANTIGEN: www.sybari.com

ANTIVIR (E+BEDV):

www.antivir.de

ANTIVIR PERSONAL EDITION

CLASSIC: www.free-av.com

AVAST ANTIVIRUS:

www.avast.com

AVG ANTIVIRUS:

www.grisoft.com

BITDEFENDER:

www.bitdefender-es.com

CLAMAV: www.clamav.net

KASPERSKY ANTIVIRUS (AVP):

www.avp-es.com

MCAFFEE VIRUSSCAN ANTI-

VIRUS: www.mcafee.com

NOD32 ANTIVIRUS SYSTEM:

www.nod32-es.com

NORTON ANTIVIRUS:

www.symantec.com

PANDA ANTIVIRUS:

www.pandasoftware.es

PC-CILLIN:

www.trendmicro.com

PER ANTIVIRUS:

www.persystems.net

SOLO ANTIVIRUS:

www.antivirus-download.com

SOPHOS ANTIVIRUS:

www.sophos.com

THE HACKER ANTIVIRUS:

www.hacksoft.com.pe

VEXIRA ANTIVIRUS:

www.centralcommand.com

VIROBOT EXPERT:

www.haurilatin.com

Páginas útiles:

- Utilidad gratis para remover spyware en línea
www.trendmicro.com/spyware-scan/
- Portal de Noticias CyberNews.Info ¡Noticias, Tecnología e Internet!
www.cibernews.info
- Ebook's
ebooks.cibernews.info

Contenido:

Tutorial de CSS	2
Tipos de Hojas	3
Unidades de Medida	7
Propiedades	9
Web: b2evolution	18
Photoshop: Bordes de pixeltool	19
Software: PC Inspector File Recovery	20

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

Una hoja de estilo es un conjunto de reglas que se aplican a una o más etiquetas HTML y definen qué apariencia tendrá el contenido de esa hoja al enviarla a pantalla, al imprimirla, o con cualquier dispositivo que se use para ver una página web. El lenguaje y reglas de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada.) hacen este trabajo de forma muy simple y clara, facilitando (y mucho) la tarea de diseñar páginas web atractivas y vistosas.

Estructura

Las hojas de estilo en cascada tienen una estructura muy simple pero muy flexible y potente. Gracias a esto podemos definir la apariencia de cada elemento o grupo de ellos con total comodidad, definir distintas apariencias en función del medio por el que se mostrarán, y cambiarla posteriormente, si es necesario, de una forma muy simple y rápida.

Sintaxis

La sintaxis básica de CSS es muy simple e intuitiva. Consta de una serie de **Reglas** que describen la forma en que se visualiza cada uno de los elementos del siguiente modo:

```
Selector {Propiedad: Valor;}
```

Por ejemplo:

```
H1 {align: left;}
```

A la primera parte (fuera de los corchetes) se le llama **Selector**, y es la que indica a qué elemento HTML afectará (en este caso a las cabeceras de nivel 1).

Lo que hay entre los corchetes es lo que se denomina **Declaración**, y es donde indicamos cómo se va a ver el selector.

La declaración está formada a su vez por una **Propiedad** (en

el caso de arriba *align*, alineación) a la que se asigna un Valor (*left*, izquierda, en el ejemplo).

O sea, lo que dice la regla de arriba es que queremos que las cabeceras de primer nivel aparezcan alineadas a la izquierda.

Si quisiéramos que los párrafos aparecieran centrados y en color rojo, podríamos escribir esto:

```
P {align: center;}
```

```
P {color: #ff0000;}
```

Con la primera regla indicamos que queremos una alineación centrada y con la segunda asignamos el color rojo.

Pero este modo es poco eficiente. Si queremos especificar varias propiedades tendremos que repetir la línea muchas veces. Para simplificar esto, CSS permite unir varias declaraciones en una misma regla, de modo que lo anterior se podría escribir, de forma mucho más clara y concisa, del siguiente modo:

```
P {align: center; color: #ff0000;}
```

Que tiene exactamente el mismo significado. Fíjate bien en que las declaraciones se separan unas de otras por medio del signo "punto y coma" (;), y que cada propiedad se separa de su valor por medio del signo "dos puntos" (:). También se podría haber escrito en la forma:

```
P {
  align: center;
  color: #ff0000;
}
```

O con cualquier otra combinación de indentados. La única diferencia es la legibilidad y comodidad a la hora de editar el código.

Hay veces en que querrás asignar las mismas reglas a varios selectores distintos. Esto lo puedes hacer repitiéndolas para cada uno de ellos o de esta manera:

```
H1, H2, P {align: center; color: #ff0000;}
```

Las reglas se aplican a los tres selectores (que han de ir separados por comas) de modo que, tanto las cabeceras de primer y segundo nivel como los párrafos se mostrarán con alineación centrada y color rojo.

Algunas propiedades pueden tener varios valores. Este es el caso, por ejemplo, de la propiedad *font*, que permite asignar tipo de fuente, tamaño, color, y varias cosas más. Esto se puede hacer separando todos los valores de una misma propiedad por espacios en blanco (), sin ningún signo de puntuación, así:

```
P {font: Arial 12/14 #00ff00;}
```

La misma propiedad, *font*, permite elegir varios tipos de fuente para, en el caso de que el sistema no tenga instalado uno de ellos, recurrir al siguiente (si no se encontrase ninguno de ellos, se usaría la fuente por defecto del navegador). esto se hace separando los posibles valores por medio de comas (,):

```
P {font: arial, helvetica, sans-serif;}
```

También se pueden hacer combinaciones algo más complejas, como la siguiente:

```
H1, H2 {align: center;}
```

```
H2 {color: #ff0000;}
```

Las cabeceras de primer nivel aparecerán centradas, y las de segundo centradas y en color rojo.

Otra combinación posible es

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

asignar una regla a un selector solo si este está dentro de otro selector, por ejemplo:

```
LI B {color: #ff0000;}
```

Que asigna color rojo al elemento solo si este está dentro de un elemento . Fíjate bien en que no hay signos de puntuación separando los selectores, solo un espacio en blanco. A esto se le llama **Selectores Contextuales**, porque hacen que la apariencia dependa del contexto donde se encuentra el selector.

También es posible combinar las reglas anteriores para obtener aplicaciones más complejas de las reglas:

```
H1, LI B {color: #ff0000;}
```

Que asignaría el color a todos los elementos <H1> y a los elementos que estén dentro de un elemento .

Clases

Lo visto hasta ahora está muy bien. Ya podemos hacer, por ejemplo, que todos los párrafos de un documento aparezcan centrados. Pero ¿Y si queremos que, además, algunos aparezcan alineados a la derecha y en color azul? Pues para esto tenemos las clases.

La clase es el nombre que se le asigna a una etiqueta HTML por medio del atributo CLASS para luego poder hacer referencia a ella. Gracias las clases podemos asignar propiedades a solo una parte de los selectores de un mismo tipo. Por ejemplo, si hemos definido uno o más párrafos por medio de la etiqueta <P CLASS="azul">, podemos asignarle color azul y alineación a la derecha con la siguiente regla:

```
P.azul {align: right; color: #0000ff;}
```

Que afectará SOLO a los párrafos que tengan su en atributo CLASS el valor "azul", sin modificar los demás. La clase se especifica escribiéndola tras el selector y un punto (sin dejar espacios). Si omitimos el selector y escribimos solo un punto y el nombre de la clase, la regla se asignará a TODOS los selectores que tengan ese nombre de clase (ya sean párrafos, cabeceras, links...).

Identificadores

Los identificadores, que se asignan a una etiqueta HTML con el atributo ID, funcionan igual que las clases, pero con la diferencia de que se pueden emplear solo una vez en un documento HTML. Es decir, que si tienes un párrafo al que has identificado como <P ID="fulano">, ese es el único elemento, sea párrafo o no, que puede tener ese ID. Para asignar una regla a ese párrafo usaremos el signo "almohadilla" (#)de la siguiente forma:

```
#fulano {align: center;}
```

Fíjate bien en que no hemos puesto el selector (P en este caso). Después de todo no es necesario: Solamente puede existir un "fulano" en todo el documento. Como te estarás imaginando, los identificadores se emplean mucho menos que las clases.

Pseudoclases

Las pseudoclases son un tipo especial de clase que no definimos nosotros, si no que ya están definidas por el estándar CSS. Actualmente solo existen tres pseudoclases, y solo afectan al elemento <A>:

- A:link
- A:active
- A:visited

Que se corresponden, respectivamente, con los link que aún no han sido visitados (A:link), los que están activos, mientras el navegador busca la dirección a la que hace referencia (A:active), y los que ya han sido visitados (A:visited). Funcionan igual que las clases, pero usando dos puntos (:) para separarlos del selector.

Pseudoelementos

Los pseudoelementos son un tipo especial de clase que describen elementos particulares de la tipografía. Solo existen dos en CSS1, y los navegadores actuales los ignoran casi totalmente. Se pueden aplicar a cualquier selector de texto, se escriben igual que las pseudoclases, usando dos puntos para separarlos del selector, y son:

- SELECTOR:first-line
- SELECTOR:first-letter

(En selector debes poner el selector al que quieras aplicar el pseudoelemento en cada caso)

"first-line" se refiere a la primera línea del selector, y first-letter a la primera letra. Por ejemplo:

```
H1:first-letter {color: #ff0000;}
P{color: #00ff00;}
```

Esto asignaría un color rojo a la primera letra de las cabeceras de primer nivel y verde a la primera línea de cada párrafo.

Uso de hojas de estilo

Tipos de hojas de estilo

¿Cómo se aplican las hojas de estilo a una página web? Existen tres tipos de aplicación de estilos para el html, según en que contexto se usen. estos tipos son, por orden de especificidad, las siguientes: Reglas de estilo

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

integradas, Hojas de estilo incrustadas y Hojas de estilo externas. Las reglas integradas se aplican a un solo elemento en un documento HTML, las hojas incrustadas se aplican a todo un documento, y las hojas externas pueden aplicarse a varios documentos. Se pueden usar los tres tipos de hoja en un mismo documento, y cada uno de ellos tiene preferencia sobre los siguientes. Veámoslo más despacio:

Reglas de estilo integradas

Las reglas de estilo integradas son la forma más elemental de usar CSS. se trata simplemente de asignar una regla a una etiqueta HTML por medio de su atributo STYLE de la siguiente forma:

```
<P STYLE="color: #00ff00;">
```

Este párrafo (y solo este, sin afectar a los demás párrafos del documento) aparecerá en color rojo. Como puedes ver, es una forma muy simple y poco eficiente de usar CSS, ya que tendríamos que repetir la regla por cada uno de los párrafos que queramos mostrar en ese color. Además, para poder usar este método sin tener posibles problemas de compatibilidad, deberíamos poner, en la cabecera del documento HTML (entre las etiquetas <HEAD> y </HEAD>) la siguiente declaración:

```
<META HTTP-EQUIV="Content-Style-Type"
CONTENT="text/css">
```

Que sirve para avisar al navegador de que el lenguaje que estamos usando en las hojas de estilo es CSS.

Hojas de estilo incrustadas

Las hojas de estilo incrustadas se definen en la cabecera de un documento HTML, y se aplican a todo este. Para ello contamos con la instrucción <STYLE>...</STYLE>, que usaremos del siguiente modo:

```
<STYLE TYPE="text/css">
<!--
Reglas de estilo aquí...
...
...
-->
</STYLE>
```

El atributo TYPE de STYLE sirve, otra vez, para explicarle al navegador que el lenguaje que usaremos en la hoja de estilo va a ser CSS. Las etiquetas de comentarios (<!-- -->) se usan para evitar que los navegadores antiguos (y que, por tanto, no reconozcan la etiqueta STYLE) muestren el contenido de la hoja en pantalla. Entre las etiquetas de comentario es donde se indicarán las reglas de estilo tal y como ya las hemos visto. Un ejemplo de una hoja de estilo incrustada es este:

```
<STYLE TYPE="text/css">
<!--
IMG { vertical-align: 50%;}
P {color: #00ff00;}
A {font-family: sans-serif;}
A:link {color: #0000ee;}
A:visited {color: #551a8b;}
A:active {color: #ff0000;}
-->
</STYLE>
```

Que debería ir entre las etiquetas <HEAD> y </HEAD> de la página Web. Este tipo de hoja se aplica a todo el documento, salvo que entre en contradicción con una regla de estilo integrada

que, al tener preferencia, se impondría sobre esta. (Es decir, que si hemos definido un párrafo con la etiqueta <P STYLE="color: #000000;">, este aparecerá en color negro, aunque los demás sí tendrán color rojo. El atributo TYPE solo es necesario si no hemos definido el lenguaje de hojas de estilo con la etiqueta <META>, tal como vimos antes.

Otra cosa que puedes poner en las hojas incrustadas es la sentencia @import url("dirección de página"), donde "dirección de página" es la URL donde se encuentra un archivo con otra hoja de estilo, cuyas reglas se añadirán a las de esta. La cláusula @import debe ir siempre la primera en una hoja de estilo, antes de cualquier regla. algunos ejemplos de @import serían las siguientes ordenes:

```
@import url(hoja.css)
```

Esta declaración hace referencia a una hoja, situada en el mismo directorio que el documento, llamada hoja.css (puedes usar la extensión que quieras pero, por costumbre se suele usar la extensión *.css)

```
@import url(../hoja.css)
```

Esta otra declaración llama a la hoja "hoja.css" que se encuentra en el directorio padre de aquel en que se encuentra el documento.

```
@import url(http://host.com/
estilo/hoja.css)
```

Y esta llama a una hoja que se encuentra en el directorio "estilo" del dominio "host.com".

Hojas de estilo externas

Hojas de estilo externas son aquellas que se almacenan en un documento aparte, y a las que se llama desde el documento HTML. Obviamente, esta hoja de

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

estilo puede ser llamada desde tantos documentos como quieras. Son las más utilizadas, y permiten dotar de un aspecto uniforme a todo un conjunto de páginas. Existen dos modos de declarar las hojas externas en un documento web. El primero es exactamente el mismo que ya hemos visto con las hojas incrustadas, ya que se usa el mismo tipo de sentencia `@import url("dirección de página")`:

```
<STYLE TYPE="text/css">
<!--
@import url(dirección de página)
-->
</STYLE>
```

En realidad esto es una hoja incrustada sin reglas y con una llamada a una hoja externa. No es un método elegante ni eficiente, y algunos navegadores más antiguos pueden no reconocerlo. La mejor manera de llamar a hojas de estilo externas es mediante la etiqueta `<LINK>` de la cabecera de un documento HTML, que se usa del siguiente modo:

```
<LINK HREF="dirección de la hoja" REL="stylesheet" TYPE="text/css">
```

En `HREF` indicamos la dirección en la que se encuentra la hoja de estilo. Con `REL="stylesheet"` informamos al navegador que se trata de una hoja de estilo, en `type`, como siempre, le decimos que está escrita en lenguaje CSS. Si ya hemos usado una etiqueta `<META>` definiendo el lenguaje de estilo por defecto, podemos omitir el atributo `TYPE`. Un ejemplo de llamada a una hoja de estilo externa es el que he usado para definir el estilo de

esta misma página:

```
<LINK REL="stylesheet"
HREF="css.css" TYPE="text/css">
```

La cláusula `@import`, que ya vimos antes, puede usarse también en las hojas de estilo externas (recuerda ponerla siempre antes que cualquier otra regla).

Usando varias hojas en un solo documento

Es perfectamente posible usar varias hojas de estilo en un mismo documento. Para ello solo es necesario declararlas al inicio de este. El uso de varias hojas puede ser muy útil en muchos casos, y puede facilitarte la tarea a la hora de hacer tu site más vistoso.

La primera forma de hacer esto es la más evidente: Puedes usar una hoja externa para definir todo aquello que quieras aplicar a varios documentos, otra incrustada para cosas específicas de ese documento, y una o más reglas integradas para detalles concretos.

Otra cosa que puedes hacer es hacer que la aplicación de tus hojas de estilo dependa del modo en que vayan a visualizarse tus páginas. ¿No sería estupendo poder decirle al navegador algo del estilo de "Si lo sacas por pantalla a color usa esta hoja, si es solo texto usa esta otra, y si lo imprimes usa aquella"? Pues bien, con CSS esto es posible.

Para ello tenemos un atributo llamado `MEDIA`, que podemos usar tanto en la etiqueta `<STYLE>` como en `<LINK>`, donde especificaremos el medio concreto al que queremos dirigir la hoja. Los medios a los que nos podemos referir con `MEDIA` son los siguientes:

SCREEN: Se refiere al monitor. Es la opción por defecto. O sea, que si no pones ningún atributo `MEDIA`, la hoja se aplicará a `SCREEN`.

PRINT: Para la impresora o cualquier otro medio escrito.

PROJECTION: Para proyectores u otro tipo de dispositivos parecidos.

BRaille: Pues eso, dispositivos braille para leer con el tacto.

SPEECH: Para chismes que leen en voz alta.

ALL: Para todos los anteriores.

Hasta ahora esos son todos los dispositivos a los que se puede hacer referencia. Desafortunadamente CSS1 no contempla reglas que permitan manipular contenidos dirigidos a muchos de ellos (no sirve de nada poner `MEDIA="SPEECH"` si luego no puedes hacer que el chisme cambie el tono de voz o el volumen del sonido). Por tanto, hoy por hoy los medios que utilizarás más habitualmente son `SCREEN` y `PRINT`. Para ello puedes poner dos hojas distintas indicando en cada una cual es el medio que quieres usar:

```
<LINK REL="stylesheet"
MEDIA="screen"
HREF="pantalla.css" TYPE="text/css">
```

```
<LINK REL="stylesheet"
MEDIA="print"
HREF="impreso.css" TYPE="text/css">
```

Cuando el documento se muestre e pantalla se usará la hoja contenida en "pantalla.css", ignorando la segunda hoja. Y si se imprime se usará la hoja "impreso.css", sin hacer caso de la anterior.

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

Hojas a medida

Uno de los objetivos de CSS es hacer accesible el contenido a cualquiera, independientemente de sus circunstancias personales como podrían ser el software que use, el equipo hardware del que disponga, etc. Para ello CSS prevee que el aspecto final de un documento dependa tanto de las hojas de estilo que haya preparado el diseñador del documento como de una serie de hojas de estilo que el propio navegante tenga previstas. Esto es muy útil para el caso de que el navegante tenga algún tipo de necesidad especial que el diseñador no haya podido preveer (Yo, por ejemplo, soy miope. Quizás me pareciera necesario tener una hoja de estilo a mi medida en la que, por ejemplo, el tamaño de los caracteres sea algo mayor que el habitual). Por otro lado, quizás al diseñador le interesa que algunas cosas no puedan ser cambiadas por el navegante, pero otras sí.

Para todo ello contamos con algunas posibilidades que podrían ser muy interesantes, si los navegadores las soportaran. Desafortunadamente, los navegadores actuales no permiten estos refinamientos que vamos a ver, aunque espero que esto sea algo que se resuelva en futuras versiones.

La etiqueta `<LINK>` para declarar las hojas de estilo externas posee un atributo de que aún no hemos hablado. Se trata del atributo `NAME`, que sirve, como era de esperar, para asignarle un nombre a la hoja de estilo. Hasta ahora hemos estado usando hojas sin este atributo. Esto las convierte en las llamadas **Hojas de Estilo Persistentes**, que son aquellas que el navegante no

puede deshabilitar. Por otro lado tenemos las llamadas **Hojas de Estilo Por Defecto**, que son iguales a las anteriores, pero asignándoles un nombre con el atributo `NAME`. Estas hojas se cargan y se aplican al documento la primera vez que este se carga, pero el navegante puede decidir anularlas. Los navegadores futuros deberían tener un menú o algo parecido en el que aparecerían los nombres de las hojas y con el que se pudiera deshabilitar el uso de las que el navegante quisiera.

Por último tenemos las **Hojas de Estilo Alternativas**, que son aquellas que no se aplican al documento a menos que el navegante las seleccione. Para ello hemos de asignarle un nombre, igual que hacíamos con las hojas por defecto, pero poniendo `"ALTERNATE STYLESHEET"` en su atributo `REL`, en lugar de `STYLESHEET`, que era lo que poníamos antes:

```
<LINK REL="stylesheet"
  HREF="hoja1.css" TYPE="text/
  css">
```

```
<LINK REL="stylesheet"
  NAME="Por Defecto"
  HREF="hoja2.css" TYPE="text/
  css">
```

```
<LINK REL="alternate styles
  sheet" NAME="Alternativa"
  HREF="hoja3.css" TYPE="text/
  css">
```

La primera de estas hojas se aplica siempre, y no es posible inhabilitarla. La segunda se aplica a menos que el navegante la inhabilite. Y la tercera no se aplicará a menos que el navegante decida habilitarla expresamente.

La Cascada de CSS

Las hojas de estilo en cascada se llaman así porque pueden

usarse unas sobre otras para sumar reglas y aplicarlas todas a un mismo documento. Sin embargo, esto puede llevar a problemas de interpretación: Si hay dos hojas que proponen distintas reglas para un mismo selector ¿Cuál de ellas debe ser interpretada?

Herencia

Algunas de las características que asignemos en una hoja de estilo pueden heredarse de unos selectores a otros. La información al respecto de si una característica es heredable suele ser, generalmente, bastante intuitiva. De todos modos puede consultarse en la descripción de la propiedad. Las reglas que controlan esta herencia son muy simples: Todos los selectores que estén contenidos en otro selector heredan de este los valores que se hayan asignado a sus propiedades (si tienen esa propiedad), salvo si se especifica para ellos otro valor. Por ejemplo, la siguiente regla:

```
BODY {border: thin; color:
  #ff0000;}
```

Asigna un borde fino a la página y color rojo al texto del elemento `body`. Al ser la propiedad `color` heredable, todos los elementos contenidos en `BODY` (por ejemplo párrafos) tendrán también color rojo, a menos que creamos otra regla para el elemento `<P>` especificando otro color. La propiedad `border`, sin embargo, no es heredable, de modo que, si queremos que los párrafos también tengan un borde alrededor, tendremos que crear una regla para ello.

También hay que tener en cuenta que las propiedades que tenga por defecto un elemento de HTML (como, por ejemplo, el que el contenido del elemento

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

<TT>...</TT> use una fuente monotipo o de paso fijo) tienen preferencia sobre las propiedades heredadas.

Por otro lado, y aunque realmente no es un aspecto de herencia, todas las reglas que apliques a un selector se aplican también a todas las clases de ese mismo selector, a menos que se especifique otra cosa en esa clase. Por ejemplo:

```
P {align: right; color: #0000ff;}
```

```
P.rojo {color: #ff0000;}
```

Todos los párrafos tendrán alineación a la derecha y color azul, excepto aquellos que tengan definido CLASS="rojo", que heredarán la alineación a la derecha, pero su color será rojo.

Prioridad

Veamos ahora como organiza CSS las prioridades entre hojas de estilo que pudieran entrar en conflictos. CSS estipula cinco niveles de prioridad, de tal modo que las reglas que se correspondan con un nivel superior se impondrán sobre aquellas que pertenezcan a uno inferior. Estos niveles de prioridad son, ordenados de menor a mayor, los siguientes:

Si no se aplican propiedades a un selector, se usará el valor heredado. Si no hubiera un valor heredado, se usará el valor por defecto.

Las declaraciones marcadas con **!important** prevalecen sobre aquellas que no estén marcadas.

Todas las hojas de estilo prevalecen sobre los valores por defecto del navegador. Las hojas de estilo del diseñador prevalecen sobre las del navegante. Las reglas incrustadas se imponen so-

bre las hojas vinculadas que, a su vez, tienen más peso que las externas.

Los selectores de identificación **ID** se imponen ante los selectores de clase **CLASS**, que son más importantes que los selectores contextuales los cuales, a su vez, se imponen sobre los selectores normales.

Las reglas de una hoja que sean posteriores a otras con las que entren en conflicto prevalecerán sobre estas.

Quizás sea necesario aclarar un poco la segunda norma. Cuando construyas una regla de estilo que consideres importante, y no quieras que pueda ser anulada por otra regla u hoja de estilo, puedes marcarla con una sentencia **!important** del siguiente modo:

```
P {color: #0000ff !important;}
```

Esto hace que el color rojo de los párrafos se imponga sobre cualquier otro color para este mismo selector que se defina en cualquier otra regla u hoja de estilo.

Realmente, estas reglas no son tan complicadas como parece. Léelas un par de veces y verás como son realmente lógicas y razonables. De todos modos, lo mejor es que experimentes por ti mismo y que procures no dejar nada al azar. Y recuerda que los navegadores no son excesivamente fieles a las normas de CSS.

Unidades de Medida

CSS utiliza una serie de unidades para poder indicar cosas como tamaños, colores, etc. Vamos a ver en detalle cuales son y cómo se usan:

Unidades de tamaño

Las unidades de tamaño se

dividen en dos grupos. El primero de ellos (y el más útil) es el de las **unidades relativas**, en el que el tamaño se mide en función de los elementos del medio en el que se representa la propiedad cuyo tamaño queremos especificar. Hay tres posibles unidades relativas:

em designa el ancho de la letra m.

ex hace lo propio con la letra x.

px indica que la medida se hace en pixels.

De estas unidades la menos recomendable es el pixel, ya que puede dar lugar a errores al visualizar la página en monitores de distinta resolución. Un ejemplo del uso de estas unidades sería:

```
P {font-size: 12px; margin: 1em}
```

Esta regla daría un tamaño de letra para el elemento <P> de 12 pixels, y un margen equivalente al grosor de la letra m.

Por otro lado tenemos las llamadas unidades absolutas, en las que se define el tamaño final, y no en relación al medio en que se presenta. Estas unidades son las siguientes:

cm Centímetros.

mm Milímetros.

in Pulgadas (inches), algunos países aún no han descubierto las ventajas del Sistema Métrico Decimal. Una pulgada equivale a 2'54 cm.

pt Puntos. Antigua unidad tipográfica (del siglo XVIII), de cuando los libros se imprimían con tipos móviles de plomo. equivale a 0'351 mm. (Por si tienes algo de curiosidad histórica, te diré que este es el punto anglo-

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

americano. en el resto de Europa el punto que se usaba, llamado punto Didot, medía 0'376 mm.).

pi Picas. Igual que la anterior, equivale a 12 puntos (unos 4'2 mm.).

Este tipo de unidades es menos recomendable, a menos que vallas a usarlo para impresión en papel u otro medio parecido.

Unidades porcentuales

Las unidades porcentuales se usan en una gran variedad de contextos, y sirven para especificar que la medida que estás indicando se refiere a un porcentaje de una medida anterior o de la medida por defecto:

Se pueden usar de dos modos, el modo absoluto y el modo aditivo. en el modo absoluto solo tiene que poner el porcentaje que quieras seguido del signo "por ciento" (%) de este modo:

```
B {font-size: 120%;}
```

Con esta regla estás indicando que quieres que el tamaño de fuente en el elemento sea de un 120% del tamaño original del texto.

El modo aditivo es semejante al anterior, pero añadiendo un signo "más" (+) o un signo "menos" (-) delante del número, para indicar si el porcentaje debe añadirse o sustraerse de la medida original. Por ejemplo:

```
B {font-size: +20%;}
```

```
I {font-size: -20%;}
```

La primera regla tiene el mismo efecto que la anterior, hace que el tamaño de fuente del elemento sea un 20% mayor que el del resto del texto. La segunda hace lo contrario para el elemento <I>, en este el tamaño de fuente será un 20% más pequeño.

Unidades de color

Cuando queremos definir un color en CSS tenemos varias posibilidades. la primera, la más práctica, la única que te recomiendo y la única que soportan los navegadores actuales es usar los viejos códigos RGB (Colores aditivos) que usábamos en HTML. estos consisten en el signo "almohadilla" (#) seguido de seis dígitos hexadecimales que corresponden, respectivamente, los dos primeros a la componente roja del color, los dos siguientes a la verde, y los dos siguientes a la azul. Veámoslo con un ejemplo:

```
B.negro {color: #000000;}
```

```
B.rojo {color: #ff0000;}
```

```
B.verde {color: #00ff00;}
```

```
B.azul {color: #0000ff;}
```

```
B.amarillo {color: #ffff00;}
```

La notación hexadecimal es aquella en la que hay 16 dígitos (en lugar de los 10 de la notación decimal). estos dígitos son: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C ,D, E, F. (No importa si usas mayúsculas o minúsculas)

Existe también una paleta RGB reducida, que solo usa un dígito hexadecimal para cada color, que se usaría de este modo:

```
B.negro {color: #000;}
```

```
B.rojo {color: #f00;}
```

```
B.verde {color: #0f0;}
```

```
B.azul {color: #00f;}
```

```
B.amarillo {color: #ff0;}
```

También puedes usar el siguiente formato, muy semejante a los anteriores pero con dígitos decimales:

```
B.negro {color: rgb(0,0,0);}
```

```
B.rojo {color: rgb(255,0,0);}
```

```
B.verde {color: rgb(0,255,0);}
```

```
B.azul {color: rgb(0,0,255);}
```

```
B.amarillo {color: rgb(255,255,0);}
```

Aquí las tres componentes de color se colocan tras la palabra "rgb", dentro de paréntesis y separados por comas. Sus valores pueden ir desde 0 a 255.

Y, por último, puedes usar el mismo sistema anterior, pero indicando valores porcentuales:

```
B.negro {color: rgb(0%,0%,0%);}
```

```
B.rojo {color: rgb(100%,0%,0%);}
```

```
B.verde {color: rgb(0%,100%,0%);}
```

```
B.azul {color: rgb(0%,0%,100%);}
```

```
B.amarillo {color: rgb(100%,100%,0%);}
```

Donde los valores pueden ir desde 0'0% hasta 100'0%.

La buena noticia es que tienes nada menos que cuatro modos de definir el color entre los que escoger. La mala es que solo puedes usar el primero, o los navegadores no sabrán que hacer con ellos.

URLs

A veces será necesario indicar la URL de una imagen, un documento, o cualquier otro recurso. para ello contamos con la construcción url(dirección), que se usa del siguiente modo:

```
BODY {background: url(fondo.gif);}
```

```
P {background: url(http://host.com/fondo2.gif);}
```

Esta regla hará que la imagen "fondo.gif" se use como fondo de la página, y que la imagen "fondo2.gif" se use como fondo en los párrafos. Como puedes ver, se pueden usar tanto direcciones

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

relativas como direcciones absolutas. Las direcciones relativas se indican con respecto a donde esté ubicada la hoja de estilo, no con respecto a la página web.

Propiedades de CSS

Ha llegado el momento de ver todas y cada una de las propiedades que CSS asigna a cada elemento, junto con su sintaxis concreta y los posibles valores que puedan adoptar.

Aclaración preliminar

Antes de comenzar a ver las propiedades, voy a darte unas pequeñas notas sobre los símbolos que he usado para explicar la sintaxis de cada propiedad. En general no hay problema alguno, y he tratado de explicar todo lo necesario en el apartado "Detalles" de cada uno, pero aquí van una serie de convenciones para aclararte algo más el tema:

Las palabras escritas entre los símbolos < y >, son variables, que dependerán de cada caso, y que se explican en la sección "Detalles".

El signo | indica la disyunción "o". Si hay variables separadas por este signo, se deberá escoger una de ellas.

El signo || indica la conjunción "y/o". Si hay variables separadas por este signo, se podrán escoger una o varias de ellas.

El signo ? indica que la variable que le precede es opcional, y puede omitirse.

El signo * indica que la variable que le precede es opcional, y puede omitirse, pero que también puede aparecer repetida varias veces.

El signo (1-4) indica que la variable que le precede puede aparecer repetida de una a cuatro veces.

Los símbolos [y] se usan para agrupar variables, con objeto de simplificar la notación

Propiedades de texto

Son aquellas que modifican las características del texto, tales como el espaciado entre palabras o la alineación.

word-spacing, letter-spacing, text-decoration, vertical-align, text-transform, text-align, text-indent, line-height.

Propiedades de fuente

Son las que afectan a las características de la letra, como el tamaño o el tipo.

font-family, font-style, font-variant, font-weight, font-size, font.

Propiedades de color y fondo

Son las que definen los colores y/o imágenes que se usarán en un objeto, como el color del texto o la imagen de fondo de un párrafo.

Color, background-color, background-image, background-repeat, background-attachment, background-position, background.

Propiedades de cuadro

Estas son las propiedades que definen el modo en que se mostrará el cuadro que contiene a un elemento, ya sea este un párrafo, una tabla, imagen, etc..

margin-top, margin-bottom, margin-right, margin-left, margin, padding-top, padding-right, padding-bottom, padding-left, padding, border-top-width, border-right-width, border-bottom-width, border-left-width, border-width, border-color, border-style, border-top, border-right, border-bottom, border-left, border, width, height, float, clear.

Propiedades de clasificación

Son aquellas que afectan a la forma de representar los elementos de clasificación como las listas.

Display, white-space, list-style-type, list-style-image, list-style-position, list-style.

Propiedades en Detalle

Propiedades de Texto

word-spacing

Indica el espaciado entre palabras que se asignará a un texto. Usualmente no es necesario cambiarlo, pues los navegadores lo distribuyen correctamente.

Sintaxis:

word-spacing: <valor>

Detalles:

<valor> puede ser una de las siguientes opciones:

normal: Que es el valor por defecto.

X: Donde "X" es el espacio entre palabras en alguna de las unidades de CSS. Si se asigna un valor negativo, las palabras aparecerán unas sobre otras.

Valor por defecto: normal

Se aplica a todos los elementos.

HEREDABLE

letter-spacing

Sirve para asignar el espaciado entre caracteres de un texto. Usualmente, al igual que ocurría con word-spacing, no suele ser necesario cambiarlo, pues los navegadores lo distribuyen correctamente.

Sintaxis:

letter-spacing: <valor>

Detalles:

<espacio> puede ser una de

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

las siguientes opciones:

normal: Que es el valor por defecto.

X: Donde "X" es el espacio entre letras, indicado en alguna de las unidades de CSS.

Valor por defecto: normal

Se aplica a todos los elementos.

HEREDABLE

text-decoration

Aplica algunos "adornos" al texto, como subrayados o tachados.

Sintaxis:

text-decoration: <valor>

Detalles:

<valor> puede ser uno de los siguientes:

none: Indica que no se aplicará ninguna decoración.

underline: Subrayado.

overline: Una línea por encima del texto

line-through: Tachado.

blink: Texto parpadeante.

Valor por defecto: none

Se aplica a todos los elementos.

NO HEREDABLE

vertical-align

Indica la alineación vertical de los elementos de línea (son elementos de línea aquellos que no tienen un salto de carro como, por ejemplo, o).

Sintaxis:

vertical-align: <valor>

Detalles:

<valor> puede ser uno de los siguientes:

baseline: Alinea la base del elemento con la base del padre.

sub: Alinea la línea superior del elemento con la inferior del padre.

super: Alinea la línea inferior del elemento con la superior del padre.

top: Alinea la línea superior del elemento con la del padre.

text-top: Alinea la línea superior del elemento con la parte superior de la fuente del padre.

middle: Alinea el centro del elemento con el centro del padre.

bottom: Alinea la línea inferior del elemento con la del padre.

text-bottom: Alinea la base del elemento con la base de la fuente del padre.

x%: Donde x es el porcentaje de altura a la que se alineará el elemento.

Valor por defecto: baseline

Se aplica a los elementos de línea.

NO HEREDABLE

text-transform

Modifica la propiedad de mayúsculas o minúsculas del elemento.

Sintaxis:

text-transform <valor>

Detalles:

<valor> puede ser uno de los siguientes:

none: lo deja como está.

capitalize: pone en mayúsculas la primera letra de cada palabra.

uppercase: pone en mayúsculas todas las letras.

lowercase: pone en minúsculas todas las letras.

Valor por defecto: none

Se aplica a todos los elementos.

HEREDABLE

text-align

Establece la alineación del elemento.

Sintaxis:

text-align <valor>

Detalles:

left: Alinea el elemento a la izquierda.

right: Alinea el elemento a la derecha.

center: Centra el elemento.

justify: Alinea el elemento a ambos lados (lo justifica).

Valor por defecto: el predeterminado por el browser (habitualmente left).

Se aplica a los elementos de bloque.

HEREDABLE

text-indent

Indica la indentación o sangría que se hará a la primera línea de un elemento de bloque.

Sintaxis:

text-indent: <valor>

Detalles:

<valor> indica el tamaño de la sangría en alguna de las unidades CSS, ya sean absolutas o relativas.

Valor por defecto: 0

Se aplica a los elementos de bloque.

HEREDABLE

line-height

Sirve para indicar la altura de la línea de texto, sin variar el tamaño de la fuente (es decir, a una mayor altura, las líneas de texto estarán más separadas).

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

Sintaxis:

line-height <valor>

Detalles:

<valor> puede ser uno de los siguientes:

normal: sitúa la altura a su valor por defecto.

x: Donde x es un número por el que se multiplicará la altura de la fuente para obtener la altura de la línea.

x: Donde x es la altura de la línea indicada en alguna de las unidades CSS, tanto relativas (al tamaño del texto) como absolutas.

Valor por defecto: normal

Se aplica a todos los elementos.

HEREDABLE

Propiedades de Fuente

font-family

Con esta propiedad puedes escoger la fuente (tipo de letra) que quieres que se use en el selector. Para que funcione esta debe estar instalada en el ordenador del navegante. puedes escoger una fuente concreta (por ejemplo "Helvetica") o una familia de fuentes (sans-serif). también puedes usar varios nombres seguidos: El navegador tratará de usar la primera de ellas, si no está instalada lo intentará con la siguiente, etc..

Sintaxis:

font-family: [[<fuente> | <familia>],]* [<fuente> | <familia>]

Detalles:

<fuente> es el nombre de fuente que quieras usar. Si el nombre contiene espacios, deberá ir entre comillas.

<familia>: Es el nombre ge-

nérico de la familia. Sus posibles valores son:

serif: se refiere a las fuentes "con remate" como, por ejemplo, la Times.

sans-serif: Son las llamadas fuentes "de palo seco", como la Helvetica.

cursive: Es la tipografía caligráfica, como la comic

fantasy: Es un término un tanto amplio, que abarca a todas las llamadas tipografías "ornamentales"

monospace: Es el tipo de fuente monoespaciada, como e las máquinas de escribir.

Valor por defecto: (Predeterminado del navegador, usualmente Times).

Se aplica a todos los elementos.

HEREDABLE

font-style

Sirve para decidir el estilo de la fuente

Sintaxis:

font-style: <estilo>

Detalles:

<estilo> puede tener uno de estos tres valores:

normal: Es exactamente eso, lo normal.

italic: Itálica, cursiva.

oblique: Inclínada, un estilo parecido a algo intermedio entre la cursiva y la normal.

Valor por defecto: normal.

Se aplica a todos los elementos.

HEREDABLE

font-variant

Este atributo se usa, actualmente, solo para definir si quie-

res usar las mayúsculas y minúsculas de la forma tradicional o como "versalitas", es decir, usando unas mayúsculas de menor tamaño en lugar de minúsculas.

Sintaxis:

font-variant: <modo>

Detalles:

<modo> puede tener uno de estos dos valores:

normal: Es, como siempre, normal. En este caso con las minúsculas de siempre.

SMALL-CAPS: Es en la forma "versalitas".

Valor por defecto: normal.

Se aplica a todos los elementos.

HEREDABLE

font-weight

Esta propiedad sirve para escoger el "peso" de la tipografía, es decir, lo "gordas" que serán las letras. Depende de las fuentes que haya instaladas en el sistema del navegante.

Sintaxis:

font-weight: <peso>

Detalles:

<peso> puede ser uno de estos:

normal: Pues eso, normal.

bold: Negrita.

bolder: Un poco más pesada que el valor heredado. (Varios bolder acumulados incrementan el peso proporcionalmente).

lighter: Un poco más ligera que el valor heredado. (Varios lighter acumulados disminuyen el peso proporcionalmente).

100, 200, 300, 400, 500, 600, 700, 800, 900: Cada uno de estos números equivale a pesos pro-

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

porcionales. (100 es el más ligero y 900 el más pesado).

Valor por defecto: normal.

Se aplica a todos los elementos.

HEREDABLE

font-size

Sirve para especificar el tamaño de fuente que se usará.

Sintaxis:

font-size: <absoluto> | <relativo> | <tamaño> | <porcentaje>

Detalles:

:<absoluto>: Especifica un tamaño de entre uno de los siguientes (de menor a mayor):

xx-small

x-small

small

medium

large

x-large

xx-large

<relativo> modifica el tamaño heredado en un nivel:

larger: Aumenta un nivel el tamaño de letra.

smaller: Decrementa el valor heredado en un nivel.

<tamaño>: Especifica el tamaño exacto del tipo de letra en pixels, picas, u otra unidad CSS.

:<porcentaje>: Especifica el tamaño del tipo de letra en porcentaje respecto al valor heredado.

Valor por defecto: medium.

Se aplica a todos los elementos.

HEREDABLE

font

En Font se aplican todas las

propiedades anteriores. Es una especie de "propiedad resumen" en la que se pueden modificar todos los parámetros que ya hemos visto.

Sintaxis:

font:[<font-style> || <font-variant> || <font-weight>]? <font-size> [/ <line-height>]? <font-family>

Detalles:

<font-style>: Equivale a la propiedad Font Style

<font-variant>: Equivale a la propiedad Font Variant

<font-weight>: Equivale a la propiedad Font Weight

<font-size>: Equivale a la propiedad Font Size

<line-height>: Equivale a la propiedad Line Height

<font-family>: Equivale a la propiedad Font Family

Valor por defecto: El correspondiente a cada propiedad.

Se aplica a todos los elementos.

HEREDABLE

Propiedades de Color y Fondo

color

Indica el color del elemento al que se aplica.

Sintaxis:

color: <color>

Detalles:

<color> Indica el color en alguna de las unidades de color de CSS

Valor por defecto: Asignado por el Browser a cada elemento (Por ejemplo, negro para el texto.)

Se aplica a todos los elementos.

HEREDABLE

background-color

Asigna el color de fondo del elemento al que se aplica.

Sintaxis:

background-color: <color>

Detalles:

<color> Indica el color en alguna de las unidades de color de CSS.

transparent Usado en lugar de un código de color, asigna el color transparente.

Valor por defecto: transparent.

Se aplica a todos los elementos.

NO HEREDABLE

background-image

Asigna una imagen de fondo al elemento al que se aplica.

Sintaxis:

background-image: <url>

Detalles:

<url> Una URL en el formato CSS.

Valor por defecto:

Se aplica a todos los elementos.

NO HEREDABLE

background-repeat

Indica, en el caso de que una imagen de fondo sea menor que el área a cubrir, la forma en que se repetirá esta imagen.

Sintaxis:

background-repeat: <modo>

Detalles:

<modo> Indica la forma en que se repetirá la imagen, puede ser uno de los siguientes:

repeat: La imagen se repetirá tanto en horizontal como en

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

vertical.

repeat-x: La imagen se repetirá solo en horizontal.

repeat-y: La imagen se repetirá solo en vertical.

no-repeat: La imagen no se repetirá

Valor por defecto: repeat

Se aplica a todos los elementos.

NO HEREDABLE

background-attachment

Indica si una imagen de fondo permanecerá fija al hacer un scroll del contenido del elemento al que se aplica o no.

Sintaxis:

background-attachment: <modo>

Detalles:

<modo> Puede ser uno de los dos siguientes valores:

scroll: La imagen se desplazará al hacer un scroll.

fixed: La imagen no se desplazará al hacer un scroll.

Valor por defecto: fixed

Se aplica a todos los elementos.

NO HEREDABLE

background-position

Define la posición en la que se insertará una imagen de fondo.

Sintaxis:

background-position: <horizontal> || <vertical>

Detalles:

<horizontal> puede adoptar una de las siguientes formas:

x%: Donde "x" es un porcentaje que representa la posición horizontal de la imagen con res-

pecto al ancho del objeto al que se aplica.

x: Donde "x" es la distancia, en alguna de las unidades CSS, al borde izquierdo del objeto al que se aplica.

left: La imagen se pegará a la izquierda.

center: La imagen se colocará en el centro.

right: La imagen se colocará a la derecha.

<vertical> puede adoptar una de las siguientes formas:

x%: Donde "x" es un porcentaje que representa la posición vertical de la imagen con respecto a la altura del objeto al que se aplica.

x: Donde "x" es la distancia, en alguna de las unidades CSS, al borde superior del objeto al que se aplica.

top: La imagen se colocará en el borde superior.

center: La imagen se colocará en el centro.

bottom: La imagen se colocará abajo.

Valor por defecto: 0% 0%

Se aplica a los elementos de bloque y a los elementos IMG, INPUT, OBJECT, TEXTAREA y SELECT

NO HEREDABLE

background

Esta propiedad es la suma de background-color, background-image, background-repeat, background-attachment y background-position

Sintaxis:

background: <background-color> || <background-image> || <background-repeat> || <background-attachment> ||

<background-position>

Detalles:

En cada una de las variables se usa la sintaxis definida para cada una de ellas.

Valor por defecto: NINGUNO.

Se aplica a todos los elementos.

NO HEREDABLE

Propiedades de Cuadro

margin-top

Define el tamaño del margen superior de un elemento.

Sintaxis:

margin-top: <valor>

Detalles:

<valor> puede ser un porcentaje o un valor absoluto.

Valor por defecto: 0

Se aplica a todos los elementos.

NO HEREDABLE

margin-bottom

Define el tamaño del margen inferior de un elemento.

Sintaxis:

margin-bottom: <valor>

Detalles:

<valor> puede ser un porcentaje o un valor absoluto.

Valor por defecto: 0

Se aplica a todos los elementos.

NO HEREDABLE

margin-right

Define el tamaño del margen derecho de un elemento.

Sintaxis:

margin-right: <valor>

Detalles:

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

<valor> puede ser un porcentaje o un valor absoluto.

Valor por defecto: 0

Se aplica a todos los elementos.

NO HEREDABLE

margin-left

Define el tamaño del margen izquierdo de un elemento.

Sintaxis:

margin-left: <valor>

Detalles:

<valor> puede ser un porcentaje o un valor absoluto.

Valor por defecto: 0

Se aplica a todos los elementos.

NO HEREDABLE

margin

Permite definir todos los márgenes de un elemento.

Sintaxis:

margin: <valor>(1-4)

Detalles:

<valor> puede ser un porcentaje o un valor absoluto. Se pueden poner de uno a cuatro valores, separados por comas:

Un solo valor: Se aplica a todos los lados.

Dos valores: El primero se aplica arriba y abajo y el segundo a izquierda y derecha.

Tres valores: El primero arriba, el segundo a la derecha y a la izquierda y el tercero abajo.

Cuatro valores: El primero arriba, el segundo a la derecha, el tercero abajo y el cuarto a la izquierda.

Valor por defecto: Ninguno

Se aplica a todos los elementos.

NO HEREDABLE

padding-top

Define el espacio entre el borde superior de un elemento y su contenido.

Sintaxis:

padding-top: <valor>

Detalles:

<valor> puede ser un valor absoluto o un porcentaje.

Valor por defecto: 0

Se aplica a todos los elementos.

NO HEREDABLE

padding-right

Define el espacio entre el borde derecho de un elemento y su contenido.

Sintaxis:

padding-right: <valor>

Detalles:

<valor> puede ser un valor absoluto o un porcentaje.

Valor por defecto: 0

Se aplica a todos los elementos.

NO HEREDABLE

padding-botton

Define el espacio entre el borde inferior de un elemento y su contenido.

Sintaxis:

padding-botton: <valor>

Detalles:

<valor> puede ser un valor absoluto o un porcentaje.

Valor por defecto: 0

Se aplica a todos los elementos.

NO HEREDABLE

padding-left

Define el espacio entre el borde izquierdo de un elemento y su contenido.

Sintaxis:

padding-left: <valor>

Detalles:

<valor> puede ser un valor absoluto o un porcentaje.

Valor por defecto: 0

Se aplica a todos los elementos.

NO HEREDABLE

padding

Define el espacio entre los bordes de un elemento y su contenido.

Sintaxis:

padding: <valor> (1-4)

Detalles:

<valor> puede ser un valor absoluto o un porcentaje. Se pueden poner de uno a cuatro valores, separados por comas:

Un solo valor: Se aplica a todos los lados.

Dos valores: El primero se aplica arriba y abajo y el segundo a izquierda y derecha.

Tres valores: El primero arriba, el segundo a la derecha y a la izquierda y el tercero abajo.

Cuatro valores: El primero arriba, el segundo a la derecha, el tercero abajo y el cuarto a la izquierda.

Valor por defecto: 0

Se aplica a todos los elementos.

NO HEREDABLE

border-top-width

Indica el grosor del borde superior de un contenedor.

Sintaxis:

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

<p>border-top-width: <valor> Detalles: <valor> puede ser un valor absoluto o uno de los siguientes: thin: Borde fino. medium: Borde medio. thick: Borde grueso. Valor por defecto: medium Se aplica a todos los elementos. NO HEREDABLE border-right-width Indica el grosor del borde derecho de un contenedor. Sintaxis: border-right-width: <valor> Detalles: <valor> puede ser un valor absoluto o uno de los siguientes: thin: Borde fino. medium: Borde medio. thick: Borde grueso. Valor por defecto: medium Se aplica a todos los elementos. NO HEREDABLE border-bottom-width Indica el grosor del borde inferior de un contenedor. Sintaxis: border-bottom-width: <valor> Detalles: <valor> puede ser un valor absoluto o uno de los siguientes: thin: Borde fino. medium: Borde medio. thick: Borde grueso. Valor por defecto: medium Se aplica a todos los elementos.</p>	<p>NO HEREDABLE border-left-width Indica el grosor del borde izquierdo de un contenedor. Sintaxis: border-left-width: <valor> Detalles: <valor> puede ser un valor absoluto o uno de los siguientes: thin: Borde fino. medium: Borde medio. thick: Borde grueso. Valor por defecto: medium Se aplica a todos los elementos. NO HEREDABLE border-width Indica el grosor de los bordes de un contenedor. Sintaxis: border-width: <valor> (1-4) Detalles: <valor> puede ser un valor absoluto o uno de los siguientes: thin: Borde fino. medium: Borde medio. thick: Borde grueso. Se pueden poner de uno a cuatro valores, separados por comas: Un solo valor: Se aplica a todos los lados. Dos valores: El primero se aplica arriba y abajo y el segundo a izquierda y derecha. Tres valores: El primero arriba, el segundo a la derecha y a la izquierda y el tercero abajo. Cuatro valores: El primero arriba, el segundo a la derecha, el tercero abajo y el cuarto a la izquierda.</p>	<p>Valor por defecto: medium Se aplica a todos los elementos. NO HEREDABLE border-color Define el color del borde de un contenedor. Sintaxis: border-color: <valor> (1-4) Detalles: <valor> es un valor de color. Se pueden poner de uno a cuatro valores, separados por comas: Un solo valor: Se aplica a todos los lados. Dos valores: El primero se aplica arriba y abajo y el segundo a izquierda y derecha. Tres valores: El primero arriba, el segundo a la derecha y a la izquierda y el tercero abajo. Cuatro valores: El primero arriba, el segundo a la derecha, el tercero abajo y el cuarto a la izquierda. Valor por defecto: El valor de la propiedad color Se aplica a todos los elementos. NO HEREDABLE border-style Marca el aspecto que tendrá el borde de un contenedor Sintaxis: border-style: <valor> (1-4) Detalles: <valor> puede ser uno de los siguientes: none: dotted: dashed: solid: double:</p>
--	---	--

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

groove:

ridge:

inset:

outset:

Se pueden poner de uno a cuatro valores, separados por comas:

Un solo valor: Se aplica a todos los lados.

Dos valores: El primero se aplica arriba y abajo y el segundo a izquierda y derecha.

Tres valores: El primero arriba, el segundo a la derecha y a la izquierda y el tercero abajo.

Cuatro valores: El primero arriba, el segundo a la derecha, el tercero abajo y el cuarto a la izquierda.

Valor por defecto: none

Se aplica a todos los elementos.

NO HEREDABLE

border-top

Esta propiedad es, en realidad, la unión de "border-top-width", "border-style" y "border-color", permitiendo así modificar los valores de ancho, estilo y color del borde superior de un elemento.

Sintaxis:

```
border-top: <border-top-width> || <border-style> || <color>
```

Detalles:

Se puede poner una o más de las propiedades indicadas. Cada una de las opciones puede tomar los valores de la respectiva propiedad.

Valor por defecto: Ninguno

Se aplica a todos los elementos.

NO HEREDABLE

border-right

Esta propiedad es, en realidad, la unión de "border-top-right", "border-style" y "border-color", permitiendo así modificar los valores de ancho, estilo y color del borde derecho de un elemento.

Sintaxis:

```
border-top: <border-right-width> || <border-style> || <color>
```

Detalles:

Se puede poner una o más de las propiedades indicadas. Cada una de las opciones puede tomar los valores de la respectiva propiedad.

Valor por defecto: Ninguno

Se aplica a todos los elementos.

NO HEREDABLE

border-bottom

Esta propiedad es, en realidad, la unión de "border-botton-width", "border-style" y "border-color", permitiendo así modificar los valores de ancho, estilo y color del borde inferior de un elemento.

Sintaxis:

```
border-botton: <border-botton-width> || <border-style> || <color>
```

Detalles:

Se puede poner una o más de las propiedades indicadas. Cada una de las opciones puede tomar los valores de la respectiva propiedad.

Valor por defecto: Ninguno

Se aplica a todos los elementos.

NO HEREDABLE

border-left

Esta propiedad es, en realidad, la unión de "border-left-width", "border-style" y "border-color", permitiendo así modificar los valores de ancho, estilo y color del borde izquierdo de un elemento.

Sintaxis:

```
border-top: <border-left-width> || <border-style> || <color>
```

Detalles:

Se puede poner una o más de las propiedades indicadas. Cada una de las opciones puede tomar los valores de la respectiva propiedad.

Valor por defecto: Ninguno

Se aplica a todos los elementos.

NO HEREDABLE

border

Esta propiedad es, en realidad, la unión de "border-width", "border-style" y "border-color", permitiendo así modificar los valores de ancho, estilo y color de los bordes de un elemento.

Sintaxis:

```
border: <border-width> || <border-style> || <color>
```

Detalles:

Se puede poner una o más de las propiedades indicadas. Cada una de las opciones puede tomar los valores de la respectiva propiedad.

Valor por defecto: Ninguno

Se aplica a todos los elementos.

NO HEREDABLE

width

Especifica el ancho de un elemento.

Sintaxis:

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

whitdt: <valor>

Detalles:

Valor puede ser una unidad absoluta, un porcentaje, o la palabra "auto", que dejará el valor por defecto del elemento (en el caso de una imagen, por ejemplo, su ancho original).

Valor por defecto: auto

Se aplica a los elementos de bloque y a algunos elementos especiales de HTML, que tienen un ancho intrínseco, como IMG, INPUT, TEXTAREA, SELECT y OBJECT.

NO HEREDABLE

height

Especifica la altura de un elemento.

Sintaxis:

height: <valor>

Detalles:

Valor puede ser una unidad absoluta o la palabra "auto", que dejará el valor por defecto del elemento (en el caso de una imagen, por ejemplo, su altura original).

Valor por defecto: auto

Se aplica a los elementos de bloque y a algunos elementos especiales de HTML, que tienen una altura intrínseca, como IMG, INPUT, TEXTAREA, SELECT y OBJECT.

NO HEREDABLE

float

Hace que el elemento se alinee a izquierda o derecha.

Sintaxis:

float: <valor>

Detalles:

Valor puede ser uno de estos tres:

left: Que alinea el elemento

a la izquierda.

right: Que alinea el elemento a la derecha.

none: No alinea el elemento.

Valor por defecto: none

Se aplica a todos los elementos.

NO HEREDABLE

clear

Especifica si elemento al que se asigna la propiedad debe fluir en torno a otros elementos.

Sintaxis:

clear: <valor>

Detalles:

Valor puede ser uno de estos tres:

left: Permite que fluya a la izquierda de otros elementos.

right: Permite que fluya a la derecha de otros elementos.

none: No permite que fluya.

both: Permite que fluya a ambos lados de otros elementos.

Evidentemente, la posibilidad de fluir dependerá de la alineación del o los elementos en torno a los que se fluya.

Valor por defecto: none

Se aplica a todos los elementos.

NO HEREDABLE

Propiedades de Clasificación

display

Especifica la forma en que se mostrará un elemento.

Sintaxis:

display: <valor>

Detalles:

valor puede ser uno de los siguientes:

block: Como un elemento de bloque. Se añade un retorno de carro antes y después del elemento (Como, por ejemplo, en el elemento <P>).

inline: Como un elemento de línea. No se añaden retornos de carro (Como, por ejemplo, en el elemento <I>).

list-item: Igual que block, pero añadiendo una marca de lista al principio del elemento (Como, por ejemplo, en el elemento).

none: El elemento no se muestra (como en las etiquetas de comentario).

Valor por defecto: block.

Se aplica a todos los elementos.

NO HEREDABLE

white-space

Especifica el modo en que se tratarán los espacios en blanco.

Sintaxis:

white-space: <valor>

Detalles:

valor puede ser uno de los siguientes:

normal: Si hay varios espacios seguidos, se representarán como uno solo.

pre: Se representarán todos los espacios.

nowrap: La línea no se cortará al llegar al extremo de la página.

Valor por defecto: normal.

Se aplica a todos los elementos.

HEREDABLE

list-style-type

Selecciona la marca que acompaña a un elemento de lista.

Webmaster: Tutorial de CSS (Cascading Style Sheet. Hojas de Estilo en Cascada)

Sintaxis:

list-style-type: <valor>

Detalles:

valor puede ser uno de los siguientes:

disc: Un disco.

circle: Un círculo.

square: Un cuadrado.

decimal: Números (1,2,3,4...).

lower-roman: Números romanos en minúscula(i,ii,iii,iv...).

upper-roman: Números romanos en mayúscula (I,II,III,IV...).

lower-alpha: Letras en minúscula(a,b,c,d...).

upper-alpha: Letras en minúscula(A,B,C,D...).

none: Ninguno.

Valor por defecto: disc.

Se aplica a los elementos que tengan definido su atributo display como list-item.

HEREDABLE

list-style-image

Selecciona la imagen que se usará como marca de elemento de lista.

Sintaxis:

list-style-image: <valor>

Detalles:

<valor> puede ser una URL indicando la dirección del archivo gráfico que contiene la imagen que se usará como marca, o ser el valor "none", en cuyo caso se aplicará la propiedad list-style-type.

Valor por defecto: none.

Se aplica a los elementos que tengan definido su atributo display como list-item.

HEREDABLE

list-style-position

Indica si la marca de lista aparecerá dentro del cuadro al que pertenece (esto es, alineada a la izquierda con ese contenido) o si, por el contrario, aparecerá fuera del cuadro (o sea, algo más a la izquierda que dicho contenido).

Sintaxis:

list-style-position: <valor>

Detalles:

<valor> puede ser uno de estos dos:

inside: la marca aparecerá

dentro del cuadro.

outside: la marca aparecerá fuera del cuadro.

Valor por defecto: outside.

Se aplica a los elementos que tengan definido su atributo display como list-item.

HEREDABLE

list-style

Esta propiedad es la unión de "list-style-type", "list-style-position" y "list-style-image".

Sintaxis:

list-style: <list-style-type> || <list-style-position> || <list-style-image>

Detalles:

Se puede poner una o más de las propiedades indicadas. Cada una de las opciones puede tomar los valores de la respectiva propiedad.

Valor por defecto: Ninguno.

Se aplica a los elementos que tengan definido su atributo display como list-item.

HEREDABLE

Allan Psicobyte

Fuente: *Psicobyte.Com*

Web: **b2evolution**

B2evolution es uno de los mejores gestores de contenidos que puedes encontrar en la red. Te permite tener múltiples bitácoras con: múltiples idiomas y múltiples usuarios. Está disponible también en Español.

Se puede ejecutar en casi cualquier proveedor (PHP y MySQL).

La instalación es realmente

muy fácil, y lo es mucho más si tienes a tu disposición el *Fantástico* en tu *CPANEL*.

Una gran característica que posee es que puedes actualizar desde otro gestor de contenidos, la lista incluye a : Cafelog/b2, GeekLog, MiniBlog y Movable Type.

A través de la página oficial puedes encontrar themes para

darle un tono más vistoso a tu nuevo gestor. Adicionalmente, tienes a tu disposición un manual en línea donde podrás hallar toda la información que necesites. Descárgalo desde: <http://b2evolution.net>

¡Éxitos!



Algunos me han preguntado como hago los bordes en mis trabajos. Bien, veremos aquí una manera de crear un borde interesante para nuestro arte, muy al estilo 'pixeltool'.

Primer paso: Una imagen de papel envejecido. Este que aquí utilizo lo encontré en la muy útil



galería de Temabina Stock [1]. Ajusto un poco el tamaño, y mi imagen está listo para comenzar a trabajar. Debo re-

servar aquí algo de espacio para darle un aspecto más interesante al borde. Cuando Ya estoy conforme con la imagen, aplico 'Merge down'. Esto transformará mi layer en en un fondo, sin capas.

Ahora añado una nueva textura, nuevamente de Temabina Stock [1]. Utilizaré aquí sólo una



porción de la imagen. Con una 'layer mask' deo sólo lo que me interesa de la imagen, utilizando para esto una corta

degradación hacia el lado derecho. Este layer (layer 1) debe ser en modo de fusión de capa normal.

En este punto necesito añadir más 'interés' a la imagen. En una nueva capa, añado otra textura, para resolver esto. Fácil,



esto es utilizar diferentes modos de layers, y al mismo tiempo, variados niveles de opacidad. este será mi layer 2.

Después de esto, veremos que aparte de aparecer detalles, también nuestra imagen tiende a verse algo opaca y oscura. Aquí viene otro truco: Un adjustment



layer, (hue/saturation) en modo overlay. Eso ayudará a dar algo de 'vida' a la imagen. Yo utilicé aquí estos valores: Hue: 29; saturation: 23; lightness: 0. La opción 'colorize' debe estar verificada.

Decidí incluir una textura más, en un layer entre el layer 2



¿no? este layer lo pongo en multiply mode, y algunos detalles ahora serán parte de la imagen. El secreto

detrás de esto es utilizar diferentes modos de layers, y al mismo tiempo, variados niveles de opacidad. este será mi layer 2.

layer, (hue/saturation) en modo overlay. Eso ayudará a dar algo de 'vida' a la imagen. Yo utilicé aquí estos valores:

Hue: 29; saturation: 23; lightness: 0. La opción 'colorize' debe estar verificada. Decidí incluir una textura más, en un layer entre el layer 2 y el adjustment layer. Esta vez es un detalle de una textura de hueso, de la gente de Pb Stock [2]. Este viene

a ser mi layer 3, un layer en modo multiply al 50% de opacidad.

Este es el resultado final.

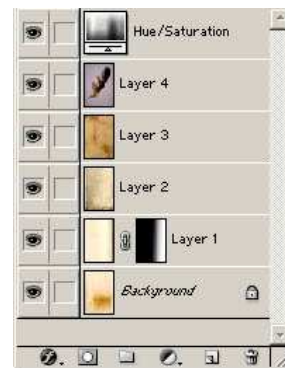


Añadí otra imagen, para mostrar con esto como se ven los bordes aplicados a una imagen. La pluma es cortesía de Pb

Stock [2]. Ese es mi layer 4, en modo overlay.

Así lucen mis layers en este ejemplo, con todo el trabajo listo.

Creo que la clave al momento



de crear cualquier trabajo artístico, reside en la imaginación. La técnica es muy importante, pero puedes aprender a usar las herramientas de muchas maneras, pero tu imaginación es única y no tiene precio.

Si quieres ver más de mis trabajos, dale un vistazo a mi sección de arte o mi galería en deviantART [3].

¡Paz y Dios les Bendiga!

Tutorial por Alonso Espinoza

- [1] <http://temabinastock.deviantart.com/>
- [2] <http://pbstock.deviantart.com/>
- [3] <http://pixeltool.deviantart.com/>

Fuente: *Pixeltool.com.ve.*



DkX Network

darkis.x@gmail.com

www.darkisx.com

D k X

Artistas. ¡Corazón, Alma y Pasión en Pro de una Meta!. "... *no dejen que las palabras los hieran ni que los halagos los cieguen, solo sean ustedes mismos y comiencen a creer no en lo que son, sino en lo que valen...* "

Estoy convencido de que la salida de Latinoamérica es por el lado del arte...

... sólo los artistas van a desentrañar y poner de relieve el ser latinoamericano, de ahí es donde va a derivarse todo lo demás...

Jonuel Brigue

La Próxima Visión del Mundo es Hoy...

Software: PC Inspector File Recovery



Se trata de un programa capaz de recuperar archivos y rescatar datos perdidos en el disco duro o disquetes, de uso gratuito. Soporta los sistemas FAT 12 (disquetes), FAT 16, FAT 32 y también NTFS (NT, 2000, XP).

La versión actual 3.x, de PC Inspector File Recovery, encuentra automáticamente las unidades de disco aunque los sectores boot o la tabla de asignación de archivos (FAT) hayan sido borrados o estén defectuosos (esto no funciona en los sistemas soportados por NTFS).

También reconstruye los archivos con la hora y fecha originales, y ofrece asistencia en la tarea de guardar datos reconstruidos de unidades de red.

Reconstruye también los datos en los que no exista posible indicación del directorio a que pertenecen. Otros productos similares no son capaces de esta reconstrucción. La "Función Es-

pecial de Recuperación" soporta los siguientes formatos de archivos: ARJ AVI BMP CDR DOC DXF DBF XLS EXE GIF HLP HTML HTM JPG LZH MID MOV MP3 PDF PNG RTF TAR TIF WAV ZIP

PC Inspector File Recovery 3.x es un producto FREEWARE, es decir este software es completamente gratuito.

Algunas indicaciones

Para la utilización del PC Inspector File Recovery 3.x se requiere alguna de las versiones del sistema operativo Windows, y (muy importante), una segunda unidad de disco. ¡En ningún caso debe instalar esta herramienta en la misma unidad de disco de la que pretende rescatar o recuperar datos!

Es más, cuando borre algo accidentalmente, intente no seguir utilizando el sistema. El tiempo de uso después del borrado de archivos, así como cualquier desfragmentación del disco, disminuye proporcionalmente las posibilidades de recuperar la información perdida.

Este software se instala en

una segunda unidad de disco, desde la que se procede a su ejecución.

Recuerde sin embargo, que PC Inspector File Recovery 3.x no puede ayudarlo si los problemas existentes se deben a una falla en el soporte mecánico de los datos (daño físico). Tampoco si el disco duro no es reconocido por la BIOS, o se oyen ruidos extraños (fallas en el mecanismo físico).

En la página de PC Inspector File Recovery 3.x, se sugiere que en esos casos, usted se ponga en contacto con los expertos de "CONVAR - Die Datenretter", <http://www.convar.de/>, para solucionar estos casos. Obviamente esto último no es gratuito.

No obstante ello, "CONVAR - Die Datenretter", pone a la disposición de todos los usuarios esta herramienta en forma totalmente gratuita.

Página de descarga:

http://www.pcinspector.de/file_recovery/es/download.htm

Espero sea de provecho esta información para ti, nos vemos en un próximo PROXIÓN.