

CSS1

manual de referencia

Autor: Jorge Sánchez (www.jorgesanchez.net) año 2003

Basado en la normativa sobre CSS1 disponible en:
<http://www.w3.org/TR/1999/REC-CSS1-19990111>

Justificación de CSS

CSS es la abreviatura de *Cascade Style Sheets*, hojas de estilo en cascada. Esta tecnología pretende eliminar los problemas que tiene el HTML. HTML tiene el problema de que es difícil controlar exactamente el formato de las páginas Web. Con CSS se pretende crear documentos que especifiquen claramente como deben actuar las etiquetas de HTML.

De hecho con CSS se intenta evitar la excesiva utilización de tablas en las páginas y el uso de los llamados “trucos HTML”. En su lugar no se indica en el código HTML el estilo del documento sino que se indica lo que es cada elemento de la página. El formato en el que se muestra cada objeto es lo que se define mediante CSS.

Versiones CSS

Hay actualmente tres estándares CSS:

- 1> **CSS1.** Formatos del texto, párrafo, fondo, colores, márgenes y listas.
- 2> **CSS2.** Posicionamiento (capas, CSSP), soporte de XML, fuentes descargables, funciones de impresión, algunos comportamientos.
- 3> **CSS3.** (En borrador). Soporte para texto vertical, columnas. Comportamientos avanzados e integración con gráficos y fuentes.

Ventajas

- Se ahorra código porque cuando los estilos se repiten, no hace falta repetir todo el código
- Además se ahorra código porque los estilos creados pueden servir para todo el sitio web
- La modificación de la página es más sencilla (si se utiliza adecuadamente)
- El código HTML es mucho más sencillo

Desventajas

- No todos los navegadores lo soportan (sólo versiones 4 o superior de Explorer y Netscape lo hacen más o menos adecuadamente)
- No se sigue el estándar por todos los navegadores por lo que el resultado del CSS difiere de un navegador a otro (Opera, Konqueror y Mozilla - Netscape 6/7 son los más fieles con el estándar, Explorer implementa numerosos códigos propios)

Inserción de códigos CSS

En una etiqueta de HTML

Se puede escribir código CSS dentro de una etiqueta HTML de formato (P por ejemplo). En ese caso la acción del código se aplicará para el texto que esté dentro de

la etiqueta. Hay que hacer uso en este caso del atributo STYLE de las etiquetas de formato. Ejemplo:

```
<p style="font-family: Arial, Helvetica, sans-serif">Hola</p>
```

En este caso la palabra **Hola** aparecerá en letra Arial.

En una página HTML

Para escribir código CSS en una página HTML hay que hacer uso de la etiqueta **<style>** en la cabecera (**head**) del documento. De esta forma:

```
<style type="text/css">
código CSS
</style>
```

La etiqueta se debe poner dentro de la cabecera de la página (etiqueta HEAD). En este caso, el código así escrito afecta a todo el código HTML. Ejemplo:

```
<style type="text/css">
  p {
    color:red;
  }
</style>
```

Eso hace que todas las etiquetas p de la página web aparezcan de color rojo.

En navegadores antiguos sin soporte para CSS se ignorará la etiqueta **style** pero no el contenido de ésta, que será considerado un texto normal y se mostrará al principio de la página.

Para evitar este efecto en navegadores antiguos y hacer que se ignore el código CSS se coloca el texto CSS entre comentarios, de esta forma:

```
<style type="text/css">
<!--
  p {
    color:red;
  }
-->
</style>
```

Los navegadores modernos ignorarán el comentario y los navegadores viejos sí le harán caso y se saltarán el código.

En un archivo separado

Se puede también escribir código CSS en un archivo de texto cuya extensión debe ser CSS. Más tarde se puede vincular este archivo a la página Web que desee utilizar los estilos definidos en el archivos.

La importancia de los estilos CSS es mayor si estos se escriben en un archivo separado. De ese modo podremos aplicar los estilos definidos en el archivo a la página

que queramos. Esto proporciona una mayor homogeneidad y mayor facilidad de modificación del formato.

El archivo de los estilos deberá tener extensión CSS y sólo contendrá código CSS. Para hacer que una página determinada utilice los estilos definidos en la hoja, se usa en la parte de la cabecera (**head**) de la página:

```
<link href="ruta" rel="stylesheet" type="text/css">
```

Otra forma posible es el uso de **@import** que se incluye dentro del código CSS:

```
<style type="text/css">
@import URL("ruta");
</style>
```

Esta última posibilidad no es soportada por Netscape Navigator 4.x.

Escribir código CSS

El código en CSS se escribe usando cualquier editor de texto. Este código suele afectar a etiquetas de formato HTML (P, H1, BODY, etc.).

Para ello se utiliza este formato:

```
etiquetaHTML {
    código1deformatoCSS : valor;
    código2deformatoCSS : valor;
    ...
}
```

Ejemplo:

```
P {
    line-height:10pt;
    text-decoration:underline;
    text-align:center;
}
```

Esto se realiza para todas las etiquetas HTML que se desean modificar. Cada formato escrito debe terminar con punto y coma.

Aunque CSS no diferencia entre mayúsculas y minúsculas, es conveniente escribir en minúsculas para mayor claridad y compatibilidad.

Comentarios

Si se desea colocar un comentario dentro del código, el mismo debe estar encerrado entre los símbolos `/*` y `*/`. Ejemplo:

```
p {
    line-height:10pt;
    /*El siguiente formato hará un subrayado*/
    text-decoration:underline;
    text-align:center;}
```

Aplicar el formato CSS

Redefinir etiquetas

Mediante CSS se puede determinar el formato que tomará una etiqueta HTML en concreto. Casi todas las etiquetas pueden ser formateadas, pero hay que tener en cuenta que algunas no tomarán todos los formatos. Por ejemplo la etiqueta **** se puede redefinir, pero no se la puede dar alineación de párrafo, puesto que **strong** no es una etiqueta de párrafo.

En cualquier caso para redefinir una etiqueta se usa:

```
etiquetaHTML {...códigoCSS...}
```

Ejemplo:

```
p {  
    color:red;  
    font-size:12pt}  
}  
/* La etiqueta P ahora es rojo y con letra de 12 puntos */
```

Se pueden agrupar etiquetas en el código. Ejemplo:

```
p, li, h1, h2 {  
    color:blue;  
    font-size:12pt}  
}
```

Herencias

Hay que tener en cuenta que hay etiquetas que son *padre* de otras. Es decir etiquetas que contienen a otras. En el ejemplo:

```
<p>Arturo Herrero: <em>Los años veinte<em><p>
```

La etiqueta **p** es padre de la etiqueta **em**. Esto hace que **em** herede todo el estilo que posea **p** y además añadirá el suyo propio. Por ejemplo:

```
p {  
    color:blue;  
    font-size:12pt  
}  
em {  
    font-size:14pt;  
}
```

En el ejemplo anterior, “los años veinte” tendrán color azul y tamaño 14.

Estilos por clases

Se puede conseguir que haya más de un estilo para cada etiqueta HTML, sin tener que utilizar aplicación individual. Para conseguirlo se debe utilizar las **clases**. Las clases

dividen una determinada etiqueta en varios tipos, a cada uno de los cuales se le asigna un nombre.

El nombre de la clase se pone detrás del de la etiqueta HTML separado por un punto, en el código CSS.

```
Etiqueta.clase {...códigoCSS...}
```

Luego para aquellas etiquetas que deban utilizar esa clase, se debe usar el atributo **class** y asignarle el nombre de la clase.

```
<etiqueta ...class=nombreclase...>
```

Ejemplo:

```
<style type="text/css">
  p.class1 {font-size:18px;color:red;}
  p.class2 {font-size:18px;color:green;}
</style>
...
<p class="class1">Sale en rojo</p>
<p class="class2">Sale en verde</p>
...
```

También se pueden especificar clases genéricas. En este caso se pone el nombre de una clase tras el punto, pero no se indica por delante el nombre de ninguna etiqueta. Cualquier etiqueta que utilice esa clase quedará afectada por su código CSS. Ejemplo:

```
<style type="text/css">
  .verde{color:green;font-weight:bold}
</style>
...
<h1 class="verde">Sale en verde</p>
<p class="verde">También sale en verde</p>
```

Estilos por identificador

Es un modo alternativo al anterior (y menos recomendable). En este caso sólo sirve para especificar clases genéricas. Éstas se indican así:

```
#clase{CSS}
```

Y se utilizan con el atributo **id**, de la misma forma a la vista anteriormente con el atributo **class**. El atributo id sirve para identificar de forma única una determinada etiqueta CSS. La principal utilidad es hacer que ese código sea accesible desde JavaScript para poder realizar cambios dinámicos en la página

Se pueden usar ambas modalidades a la vez (identificador y clase), pero siempre tendrá preferencia el atributo **ID** en el caso de que ambas posibilidades se utilicen para la misma etiqueta.

Selectores

Se puede conseguir hacer dos clases distintas para la misma etiqueta, distinguiendo además en qué sitio esté la etiqueta. Por ejemplo se puede hacer que la etiqueta **P** se comporte de manera diferente si se encuentra dentro de una tabla. Ejemplo:

```
<style>
  p {color:red;}
  table p{color:green;}
</style>
...
<body>
  <p>sale en rojo</p>
  <table><tr><td>
    <p>sale verde</p>
  </td></tr></table>
```

Se puede combinar esta técnica con el uso de estilos por clases (aunque se usa muy poco). Ejemplo:

```
table p.estilo1{color:red;}
```

Clases especiales. Pseudo-clases

Hay elementos en las páginas que por su naturaleza tienen un comportamiento especial. Ese es el caso de los hipervínculos (etiqueta **A**). Se pueden poner distintos formatos según el vínculo esté normal, visitado o activo. Por lo tanto se usan de esta forma:

```
Etiqueta:modificador {códigoCSS}
```

Los modificadores posibles son:

Modificador	Etiquetas que lo usan	Significado
link	A	Formato del enlace cuando es “no visitado”.
active	A	Formato del enlace cuando es el activo (sobre el que se ha hecho clic).
visited	A	Formato del enlace visitado.
hover	A	Formato del enlace cuando el ratón está situado encima (no disponible en Netscape 4, es parte de CSS2).
first-letter	Etiquetas de formato de párrafos	Permite dar formato distinto a la primera letra de un párrafo (sólo disponible en Explorer).
first-line	Etiquetas de formato de párrafos	Permite dar formato distinto a la primera línea de un párrafo (no disponible en Netscape 4).

Los selectores se pueden combinar con las pseudo-clases (aunque no funciona muy bien en Explorer):

```
a:link img{
```

```
border:solid blue;
}
```

También se pueden combinar con las clases:

```
a.verde:hover {
  color:green;
  font-size:16pt;
}
.normal:first-letter{
  font-size:48pt;
}
```

Este último ejemplo no funciona en las versiones 4 y 5 de Explorer y Navigator.

La etiqueta span

A veces se quiere poner un formato sobre un trozo de texto que está comprendido en ninguna etiqueta, entonces para poder hacerlo se usa **span**. Ejemplo:

```
<style type="text/css">
  .clase1 {
    color:red;
  }
</style>
...
<p>En un lugar de <span class="clase1">La Mancha</span>
de cuyo nombre no quiero acordarme, no ha mucho que
vivía...</p>
```

Sólo el texto “La Mancha” saldrá de color rojo.

Categorías de Estilos CSS1

Formatos de fuente (Font)

font-size

Tamaño de la fuente en pantalla. Se puede especificar de tres maneras:

- ⦿ **En modo absoluto** (*absolute-size*). Hace referencia a tamaños predefinidos.

xx-small	Fuente muy pequeña
x-small	Fuente pequeña
small	Fuente un poco pequeña
medium	Fuente normal.
large	Fuente un poco grande
x-large	Fuente grande
xx-large	Fuente muy grande

- **En modo relativo** (*relative-size*). En este caso se aumenta o disminuye el tamaño de la letra sobre el tamaño que tenía la letra en el elemento que contiene al del estilo (elemento padre). Valores:

smaller	Más pequeña
larger	Más grande

- **Modo exacto** (*length*). En este caso se indica el tamaño de la letra con su valor número. Inmediatamente tras este número se indica la medida en la que se debe medir el número (por ejemplo 12cm). Posibles medidas son:

in	Pulgadas
cm	Centímetros
mm	Milímetros
em	Relativa a la fuente actual
ex	Altura de la letra "x" actual
pt	Puntos
pc	Picas
px	Píxeles

- **Modo porcentaje**. Se especifica el tamaño según un porcentaje. Un porcentaje de 100% indicaría tamaño normal.

font-family

Indica el tipo de letra, se pueden indicar varios tipos separados por comas (ej: *font-family: "AvantGarde Bk", Arial, Helvetica, sans-serif;*)

font-weight

Peso de la fuente (grosor). Valores: **normal** | **bold** | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900. Casi ningún navegador soporta tantos pesos. Por lo que sólo funcionan bien **normal** y **bold**

font-style

Estilo de letra. Puede ser; **normal**, **italic** (cursiva) u **oblique** (normalmente se representa igual que la anterior).

font-variant

VERSALES (SMALL-CAPS). Valores: **normal** | **small-caps** No funciona en Navigator 4.5.

font

Permite desde una sola propiedad cambiar en un solo golpe todas las anteriores. Su sintaxis es:

```
font: font-style font-variant font-weight font-size/line-height font-family;
```

El orden tiene que ser estrictamente ese, pero algunas propiedades se pueden dejar sin utilizar. Ejemplos:

```
font: italic bold 16pt "Comic Sans MS", Arial,  
Helvetica, sans-serif;  
font: italic small-caps 18pt/24pt;  
font: italic;
```

El segundo ejemplo coloca un tamaño de letra 18 y una distancia entre líneas de 24 puntos.

color

Color de la fuente. Se puede especificar el nombre estándar del color, el código hexadecimal del color (#FF00BB, por ejemplo) o usar la función RGB (RGB(230,123,32) por ejemplo) o incluso RGB con porcentaje (RGB(35%, 32%, 8%)).

Formatos de texto (Text)

word-spacing

Indica la distancia entre las palabras del texto. Usa las mismas medidas que la propiedad **font-size**. Disponible desde las versiones 6 de los navegadores Explorer y Netscape y en el navegador Opera.

letter-spacing

Indica la distancia entre las letras del texto. Usa las mismas medidas que la propiedad **font-size**. Disponible desde las versiones 6 de los navegadores Explorer y Netscape y en el navegador Opera.

text-decoration

Se indican presentaciones para el texto. Las posibilidades son:

- ⊙ **underline** (subrayado),
- ⊙ **overline** (línea arriba, no disponible en Navigator 4),
- ⊙ **line-through** (tachado)
- ⊙ **blink** (parpadeo, no disponible en Explorer ni en Konqueror)

vertical-align

Posición vertical del texto (o imagen) respecto a su contenedor. Posibilidades:

- ⊙ **baseline** (en la línea base inferior del texto)
- ⊙ **sub** (subíndice)
- ⊙ **super** (superíndice)
- ⊙ **top** (arriba respecto al elemento más alto de la línea)
- ⊙ **text-top** (en la línea superior del texto)
- ⊙ **middle** (medio respecto a la altura del texto)
- ⊙ **bottom** (abajo respecto al elemento más alto de la línea)
- ⊙ **text-bottom** (en la línea inferior del texto)

- **percentage** (porcentaje respecto al texto)

text-align

Alineación horizontal del texto. Puede ser: **left**, **right**, **center** o **justify**

text-indent

Sangría de primera línea, en píxeles puntos, etc. o en porcentaje respecto al tamaño de la letra. Puede tener valor negativo.

line-height

Distancia entre líneas

text-transform

Posibilidades:

- **capitalize** (la primera letra en Mayúsculas)
- **uppercase** (mayúsculas)
- **lowercase** (minúsculas)

Formatos de fondo (background)

background-color

Color de fondo. Indicándole por su código de color (ver propiedad color) o con la palabra **transparent** (transparente).

background-image

Define una imagen de fondo. Se indica la imagen de fondo usando esta forma: **URL**(rutaAlaImagen). La imagen de fondo se superpone al color de fondo

background-repeat

Indica cómo se debe repetir la imagen en mosaico. Posibilidades:

- **repeat** (se repite en todo el fondo, opción por defecto)
- **repeat-x** (se repite sólo en horizontal)
- **repeat-y** (se repite sólo en vertical)
- **no-repeat** (no hay repetición)

background-attachment

Indica si la imagen queda fija si movemos las barras de desplazamiento:

- **scroll** (se mueve)
- **fixed** (queda fija)

background-position

Posición inicial de la imagen escogida. Se especifican dos posiciones, la vertical y la horizontal separadas por espacio.

Posiciones verticales	Posiciones horizontales
top	left
center	center
bottom	right

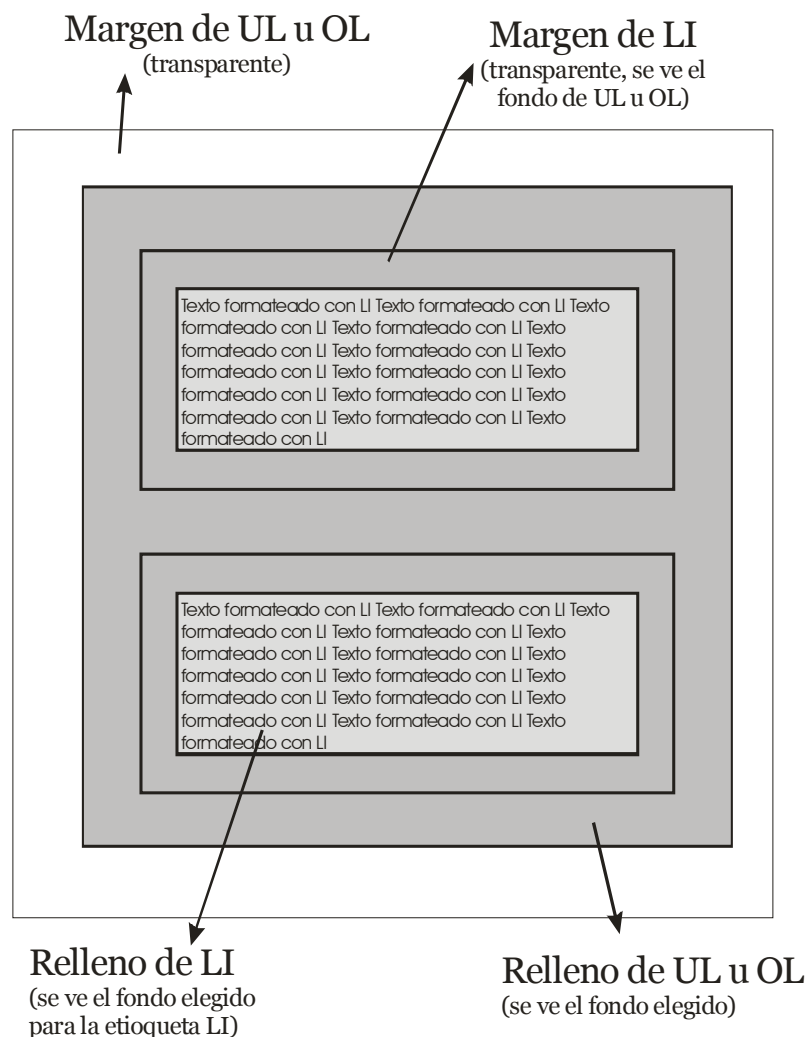
También se permite especificar la posición vertical y horizontal mediante porcentaje (sobre el tamaño del área de la ventana) o mediante píxeles (a partir del 0 0, en la esquina superior izquierda). Ejemplo *background-position: 0% 15%;*

No funciona en el Navigator 4 y Explorer puede dar problemas al usar esta propiedad sin usar *background-repeat*

background

Fija en una sola propiedad todas las propiedades de fondo. Sintaxis:

```
background: background-color background-image  
background-repeat background-attachment background-  
position
```

Resumiendo, el margen y relleno de los elementos padre, contiene al de los elementos hijos. El margen de un elemento *hijo* empieza a contar desde el relleno del *padre*.

margin

Indica la distancia del margen (puede ser negativa). Puede indicarse mediante cuatro propiedades: **margin-top** (distancia al margen superior), **margin-right** (distancia al margen derecho), **margin-bottom** (distancia al margen inferior), **margin-left** (distancia al margen izquierdo) o usando sólo la propiedad **margin** a la que daremos cuatro medidas que dispondrán del margen en el orden arriba-derecha-abajo-izquierda. Ejemplos:

```
margin-top: 10px;
margin-left: 5px;
margin-bottom: 8px;
margin-right: 6px;
```

Eso es equivalente a:

```
margin: 10px 5px 8px 6px;
```

El código:

```
margin: 20px;
```

Coloca todos los márgenes a 20 píxeles.

padding

Determina el espacio de relleno. Se usa igual que **margin** es decir: se puede usar **padding-top**, **padding-right**, **padding-bottom** y **padding-left**; o indicar todos las medidas a la vez usando la propiedad **padding**.

border-width

Especifica la anchura del borde. Se puede especificar con las propiedades **border-top-width**, **border-right-width**, **border-bottom-width** y **border-left-width**. También se puede usar **border-width** a secas y entonces indicar las cuatro medidas separadas por espacios (ver propiedad **margin**)

border-color

Especifica el color del borde. Se pueden especificar el color de los cuatro bordes con las propiedades **border-top-color**, **border-right-color**, **border-bottom-color** y **border-left-color**. También se puede usar **border-color** a secas y entonces indicar los cuatro colores separados por espacios (ver propiedad **margin**)

border-style

Se usa como las anteriores y permite indicar el estilo de línea, que puede ser:

- **dotted**. Punteada. No funciona en Explorer ni en Netscape excepto a partir de la versión 6
- **dashed**. Rayada. No funciona en Explorer ni en Netscape excepto a partir de la versión 6
- **solid**. Sólida. Línea normal, seguida.
- **double**. Línea doble.
- **groove**. Efecto de borde tridimensional hacia el interior.
- **ridge**. Efecto de borde tridimensional hacia el exterior.
- **inset**. Efecto de grabado hacia el interior de la pantalla.
- **outset**. Efecto de grabado hacia el exterior de la pantalla.
- **none**. Sin borde.

border-top, border-right, border-bottom y border-left

Permite indicar a la vez las tres propiedades de borde: anchura, estilo y color para cada línea del borde. Ejemplo:

```
border-top: 10px Yellow;  
border-bottom: 20px dashed;
```

border

Igual que la anterior pero permite indicar esas mismas tres propiedades para todos los bordes del elemento.

width

Pensada para elementos que no contengan texto (especialmente imágenes), permiten especificar una anchura (en el caso de las imágenes, independientemente del tamaño real). Si en la propiedad **height** se indica **auto**, ésta medirá lo que sea necesario de altura para que se mantengan las proporciones.

height

Permite especificar la altura del elemento. Si en la propiedad **width** se indica **auto**, ésta medirá lo que sea necesario de anchura para que se mantengan las proporciones.

float

En elementos flotantes (como imágenes por ejemplo), permite indicar como se colocará el elemento respecto al texto o imágenes que le siguen en el código. Puede ser: **left** (se colocará a la izquierda, al final del relleno de su elemento padre), **right** (a la derecha) o **none** (se coloca en la posición que le corresponde según el código).

clear

Indica si se admiten elementos flotantes alrededor. Puede ser:

- ⊙ **none**. Se permite cualquier elemento flotante (es la que se usa por defecto).
- ⊙ **left**. Se permite sólo ala izquierda.
- ⊙ **right**. Sólo ala derecha.
- ⊙ **both**. No se permite en ningún lado.

En Explorer es necesario especificar la propiedad border-style para que el borde se vea, en Navigator se debe especificar una anchura de borde por la misma razón.

Formatos de lista (list)

Se utilizan para las etiquetas de lista (**UL** y **OL**).

list-style-type

Indica qué tipo de número o viñeta se utilizará. Valores:

- ⊙ **Valores para listas de viñetas (UL)**. **disc** (disco), **square** (rectángulo), **circle** (círculo) y **none** (sin viñeta)
- ⊙ **Valores para listas numeradas (OL)**: **decimal** (número normal), **lower-roman** (números romanos en minúsculas), **upper-roman** (números romanos en mayúsculas), **lower-alpha** (letras minúsculas), **upper-alpha** (letras mayúsculas) y **none** (sin numeración)

list-style-image

Permite indicar una imagen que se utilizará como viñeta. Ejemplo:

```
ul {  
    list-style-image:URL(cuadro.gif);  
}
```

list-style-position

Indica la posición de los números o viñetas de la lista. Puede ser: **inside** (hacia dentro del relleno) u **outside** (hacia fuera, esta es la que se usa normalmente).