



ELEARNING TOTAL

Desarrollo Web HTML5 y CSS3 – Unidad 5

Diseño Web Responsive HTML5 y CSS3

Unidad 5: CSS3 Avanzado



Indice

Unidad 5: CSS3 Avanzado

CSS3 – Mayor control sobre HTML

CSS Nivel 3

Propiedades nuevas de CSS3

Bordes redondeados



Objetivos

Que el alumno logre:

- Conocer e implementar los nuevos elementos incorporados en CSS3.





CSS3 – Mayor control sobre el HTML

El objetivo inicial de CSS, separar el contenido de la forma, se cumplió ya con las primeras especificaciones del lenguaje.

Sin embargo, el objetivo de ofrecer un control total a los diseñadores sobre los elementos de la página ha sido más difícil de cubrir. Las especificaciones anteriores del lenguaje tenían muchas utilidades para aplicar estilos a las webs, pero los desarrolladores aún continuaban usando trucos diversos para conseguir efectos tan comunes o tan deseados como los bordes redondeados o el sombreado de elementos en la página.

El **nivel 1 de CSS** ya significó un avance considerable a la hora de diseñar páginas web, aportando mucho mayor control de los elementos de la página. Pero como todavía quedaron muchas otras cosas que los diseñadores deseaban hacer, pero que CSS no permitía especificar, éstos debían hacer uso de trucos para el diseño.

Lo peor de esos trucos es que muchas veces implica alterar el contenido de la página para incorporar nuevas etiquetas HTML que permitan aplicar estilos de una manera más elaborada. Dada la necesidad de cambiar el contenido, para alterar al diseño y hacer cosas que CSS no permitía, se estaban perdiendo algunos de los objetivos para los que CSS fue creado, que era el separar por completo el contenido de la forma.

El **nivel 2 de CSS** incorporó algunas novedades interesantes, que hoy ya utilizamos habitualmente, pero CSS 3 todavía avanza un poco más en la dirección, de aportar más control sobre los elementos de la página.

Así es que, la novedad más importante que aporta el **nivel 3 de CSS**, de cara a los desarrolladores webs, consiste en la incorporación de nuevos mecanismos para mantener un mayor control sobre el estilo con el que se muestran los elementos de las páginas, sin tener que recurrir a trucos o hacks, que a menudo complicaban el código de las web.



CSS Nivel 3

Con la llegada de CSS3, se agregan nuevas características a las que ya utilizábamos en CSS 2.1 para lograr los efectos y soluciones de diseño sin necesidad de requerir ayuda de otros lenguajes. Este nivel es una gran evolución para el estándar.

Entre las características más importantes encontramos atributos para manejar las sombras de los elementos (cajas y texto), bordes (colores, imágenes, redondeado), colores, opacidad, trabajo con texto (columnas), fuentes, rotación, transformación, transición y animación.

Algo que es importante aclarar, antes de comenzar a desarrollar los atributos, es que hasta la estandarización de este nivel, por una cuestión de compatibilidad, algunas propiedades requieren adicionalmente el uso de los prefijos **-moz** (para Mozilla), **-webkit** (para Chrome y Safari) y **-ms** (para Internet Explorer).

Cabe mencionar también, que si bien aplicaciones WYSIWYG (como Dreamweaver), han empezado a incorporar ayuda en el código para el uso de CSS3, tenemos también algunas herramientas online (y gratuitas!) que nos pueden ayudar a crear estilos utilizando los nuevos atributos de CSS3.

CSS3 Maker



www.css3maker.com



CSS3 Generator

CSS3 Generator

Border Radius

Border Radius: 15 px

Your Code

```
-webkit-border-radius: 15px;  
-moz-border-radius: 15px;  
border-radius: 15px;
```

Preview Area

Randy Jensen - @randyjensen - cultivatr

www.css3generator.com



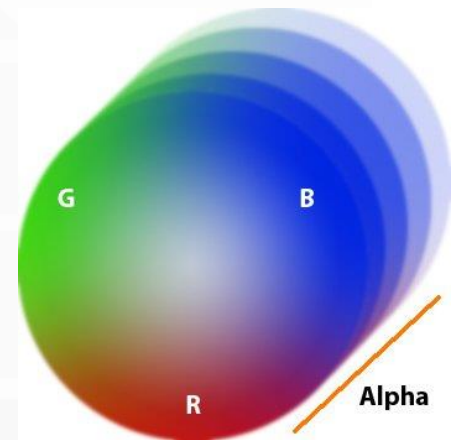
Propiedades nuevas de CSS3

Colores RGBA

Como ya fuimos viendo en unidades anteriores, los colores en HTML se expresan en valores RGB, igual que en CSS, que admite diversas notaciones para definir el color, a través de números en hexadecimal.

La notación **RGBA** es una manera de especificar colores en la que se definen cuatro valores. Los tres primeros son los bien conocidos canales RGB (rojo, verde y azul) y el cuarto parámetro es el **canal Alpha**, que no es más que el grado de transparencia u opacidad del color. El canal Alpha es un valor entre cero y uno, siendo 0 totalmente transparente y 1 totalmente opaco.

Por medio de los colores en RGBA en CSS 3, podremos aplicar nuevas transparencias a los colores que especificamos con CSS, abriendo nuevas posibilidades a los diseñadores sin necesidad de complicarse con pequeños trucos como el uso de imágenes de fondo semitransparentes en PNG, etc. Además, como los colores RGBA se pueden aplicar a cualquier elemento que soporte asignación de color, las aplicaciones aumentan todavía más.



Para definir un color RGBA, se deben especificar cuatro valores, de la siguiente manera:

- `rgba(255, 125, 0, 0.5);`

Los tres primeros valores son números en sistema decimal, que corresponden con los valores de rojo, verde y azul. Siempre tienen que ser números entre 0 y 255.

El cuarto valor es un número entre 0 y 1. Por ejemplo 0 sería totalmente transparente, 1 sería totalmente opaco y 0.5 sería una transparencia al 50%, es decir, mitad opaco mitad transparente.

Ahora veamos varios ejemplos de colores definidos con CSS y la notación RGBA.

Código CSS:

- `#transparente {
background-color: rgba(0, 0, 255, 0.1);`



}

Código HTML:

- `<div id="transparente">Esta capa tiene fondo azul, casi transparente</div>`

Código CSS:

- `.transparente2 {
 color: rgba(0,255,0,0.2);
}`

Código HTML:

- `<p class="transparente2 ">Este texto es verde es casi transparente</p>`

(ver ejemplos [transparente.html](#) y [transparente2.html](#)).

Colores HSL y HSLA

CSS3 añade además un nuevo modelo de color conocido como HSL. Estas siglas provienen del inglés *Hue*, *Saturation*, *Ligthness* o lo que es lo mismo, **tono**, **saturación y brillo**.





La sintaxis es la siguiente:

Código :

```
/*atributo: hsl(tono,saturación,brillo);*/  
background-color: hsl(360,100%,20%);
```

Teniendo en cuenta que el valor del tono puede tomar valores del 0 al 360 donde:

- 0, sería el **rojo**.
- 120, sería el **verde**.
- 240, sería el **azul**.
- 360, volvería a ser **rojo**.

Además, como en el caso del modelo RGB, al HSL se le puede añadir un **canal alpha** para definir la transparencia del color.

El resultado en código sería el siguiente, dando resultados análogos a RGBA:

Código :

```
hsla(300,130%,65%,10%);
```

Degradados

Siempre que debíamos realizar un fondo degradado para una página web, lo más común era generar una imagen que utilizábamos como patrón, que pudiese repetirse horizontal o verticalmente.

Desde la aparición del CSS3 esto ya no será necesario.

Vamos a ir haciéndolo paso a paso para que puedan ir siguiéndolo desde su editor de html (DW o simplemente el bloc de notas o cualquier editor de texto plano).

Lo primero que haremos será crear el archivo .html que contendrá nuestro degradado.

Dentro de ese archivo crearemos una declaración de estilo para agregar la propiedad del degradado



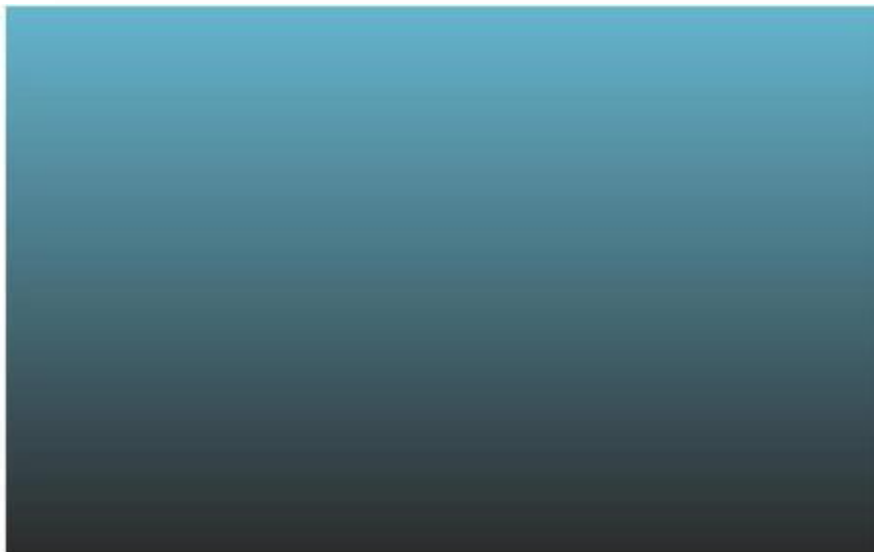
```
<style>
#contenedor {
width: 800px;
height: 500px;
margin: auto;
background: -moz-linear-gradient(top, #65b6cf, #2c2c2c);
background: -webkit-gradient(linear, 0 0, 0 100%, from(#65b6cf), to(#2c2c2c));
}
</style>
```

Vamos a ver un poco más en detalle cada sentencia:

```
background: -moz-linear-gradient(top, #65b6cf, #2c2c2c);
```

Aunque uno de los objetivos tanto de HTML5 como de CSS3 es estandarizar la visualización en los navegadores, algunos atributos aún requieren una diferenciación de código para cada navegador.

Esta es la sentencia para navegadores Mozilla Firefox, es sencillo reconocerla por su abreviatura inicial -moz, el gradiente que acabamos de realizar es el siguiente:





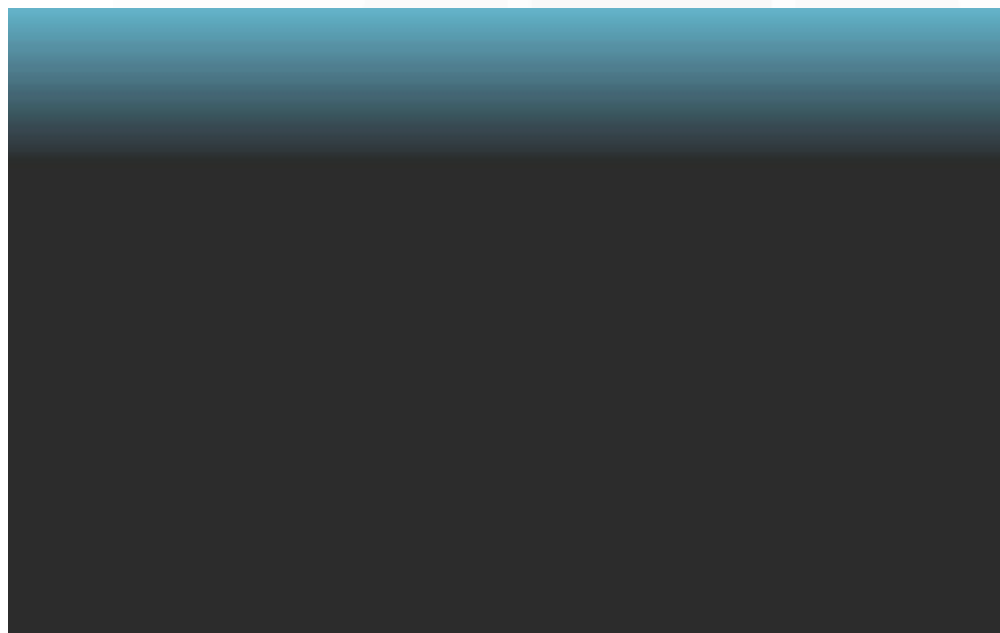
Como podemos ver, es un degradado casi por default, este reparte los colores en un 50% a cada color. Para modificar esto podemos declarar un porcentaje o tamaño a cada color, veamos entonces como explicarle al navegador desde donde empieza cada color:

- `background: -moz-linear-gradient(top, #65b6cf, #2c2c2c 100px);`

Esta es la primera forma de delimitar los colores, aquí sentenciamos que el segundo color comenzara su llenado después de 100px desde el Top. También es posible hacerlo a través de porcentajes:

- `background: -moz-linear-gradient(top, #65b6cf, #2c2c2c 25%);`

Esto se vería más o menos así:



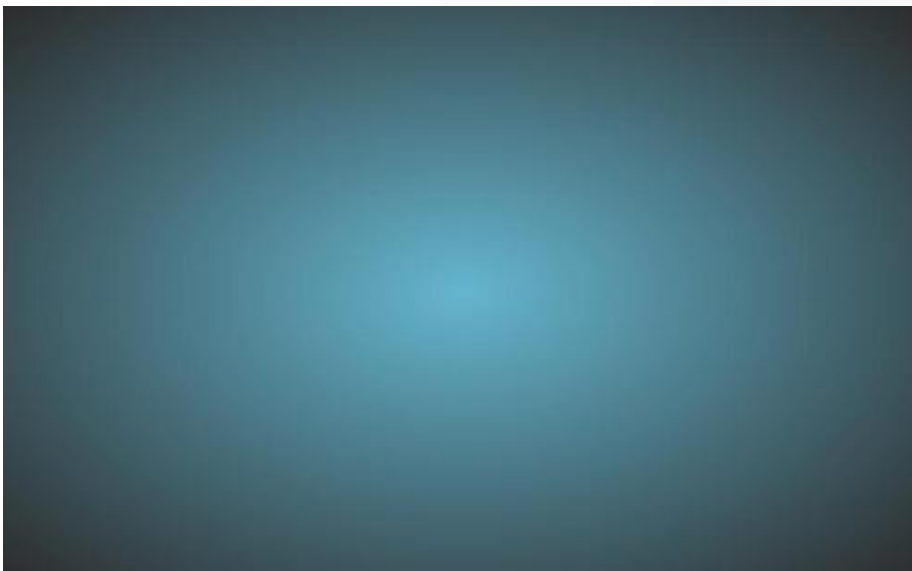
Otro elemento que podemos controlar es la cantidad de colores que van a intervenir en el degradado:

- `background: -moz-linear-gradient(top, #65b6cf, #2c2c2c 25%, #e3e3e3 80%);`



La propiedad gradiente también no permite trabajar degradados radiales:

- `background: -moz-radial-gradient(50%, #65b6cf, #2c2c2c);`





El atributo WebKit, es el que podemos visualizar desde Google Chrome (hasta el momento este atributo solo podemos verlo en Firefox y Chrome).

La sentencia que utilizamos es:

- `background: -webkit-gradient(linear, 0 0, 0 100%, from(#65b6cf), to(#2c2c2c));`

Y se interpreta de la siguiente forma:

- `linear`: indica el tipo de degradado
- `0 0`: coordenadas de inicio “eje x”, “eje y” (0 0 me indica la esquina superior izquierda)
- `0 100%`: coordenadas de finalización (0 100% indican la esquina inferior derecha)
- `from (#65b6cf)`: color inicial del degradado
- `to (#2c2c2c)`: color que finaliza el degradado.

También podemos agregarle más colores al degradado (al igual que en Firefox), con la propiedad `color-stop`:

- `background: -webkit-gradient(linear, 0 0, 0 100%, from(#dedede), color-stop(8%, white), color-stop(20%, red));`



Bordes redondeados

CSS 3 incorpora nuevas propiedades para el control de bordes de los elementos. Ahora se permiten bordes con las esquinas redondeadas, bordes con imágenes (incluso varias imágenes se pueden utilizar para definir el aspecto del borde), sombras, etc.

Tenemos la propiedad **border-radius**, que permite definir bordes redondeados en las esquinas, especificando las medidas del radio que deben darse a la curva de las esquinas.

La sintaxis de la sentencia es la siguiente:

- border-radius: 5px;

Definiría un radio de 5 píxeles en el redondeo de las esquinas del elemento. Por el momento Mozilla ha adoptado este atributo con un nombre especial, que es válido para productos como Firefox, mientras que las especificaciones de CSS3 no hayan alcanzado el estado "Candidate Recommendation", que es cuando se supone que los distintos navegadores deben implementarlas. El nombre del atributo por el momento es:

- moz-border-radius

Los navegadores basados en WebKit, como Google Chrome o Safari, también soportan las esquinas redondeadas de CSS 3, pero el atributo border-radius tampoco funciona directamente, como en el caso de Firefox, sino que hay que utilizar un "alias":

- webkit-border-radius

Por el momento, para navegadores Mozilla y WebKit que son los primeros en adaptarse a CSS3, podemos utilizar el atributo border-radius de la siguiente manera:

```
#redondo {  
border: 1px solid #553245;  
-moz-border-radius: 7px;  
-webkit-border-radius: 7px;  
padding: 10px;  
}
```



Pero el atributo border-radius tiene otras posibles configuraciones, en la que se pueden definir los valores para el radio de las cuatro esquinas por separado. De esta manera:

- moz-border-radius: 7px 27px 100px 0px;

Así estaríamos definiendo un borde redondeado con radio de 7 pixel para la esquina superior izquierda, luego 27px para la esquina superior derecha, de 100px para la inferior derecha y 0px para la inferior izquierda. (Cabe aclarar que un border-radius de 0px es un borde con esquina en ángulo recto).

Para implementar el estilo, solo debemos aplicarlo a algún elemento:

```
<div id="redondo">  
Div con bordes redondos  
</div>
```



Resumen

En esta Unidad...

Trabajamos con los nuevos elementos de CSS3.

En la próxima Unidad...

Comenzaremos a trabajar con Formularios y Multimedia HTML5.