



ELEARNING TOTAL

Programador Web / Nivel 1 – Unidad 5

Programador Web – Nivel 1

Unidad 5: Introducción a CSS3





Indice

Unidad 5: Introducción a CSS3

Bordes redondeados

Múltiples imágenes de fondo

Bordes con imágenes

Sombras

Fondos inteligentes



Objetivos

Que el alumno logre:

- Manejar los nuevos elementos incorporados a CSS3.



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>
```



MÚLTIPLES IMÁGENES DE FONDO

Una de las nuevas características de CSS 3 consiste en la posibilidad de declarar varios fondos de imagen a un elemento de la página.

Si quisiéramos lograr esto desde los estilos que conocemos hasta ahora, lo que deberíamos hacer sería crear varias etiquetas div anidadas, cada una con su imagen de fondo, algo más o menos así:

```
<div id="fondo1">
  <div id="fondo2">
    <div id="fondo3">
      contenido del elemento que va a tener 3 fondos de imagen distintos
    ...
  </div>
</div>
</div>
```

Y la declaración de estilos que da forma a esta estructura:

```
<style type="text/css">
#fondo1{
  background-image: url(fondo1.jpg);
  width: 300px;
}
#fondo2{
  background-image: url(fondo2.gif);
  background-repeat: no-repeat;
  background-position: bottom right;
}
#fondo3{
  background-image: url(fondo3.gif);
  background-repeat: no-repeat;
  background-position: center;
}
</style>
```

Los fondos se superpondrán unos a otros, siendo el fondo1 el que se vea más abajo y el fondo3 el que se verá más arriba.



Una de las nuevas características de CSS 3 consiste en la posibilidad de declarar varios fondos de imagen a un elemento de la página. Esto que hemos realizado creando varios elementos anidados y colocando un fondo en cada uno, se puede hacer en CSS 3 con un solo elemento, al que aplicaremos varios fondos distintos.

El HTML del ejemplo de varias imágenes de fondo sería el siguiente:

```
<div id="fondos">  
  texto de un único elemento  
  ...  
</div>
```

Ahora veamos el CSS 3 válido para este ejemplo:

```
<style type="text/css">  
#fondos{  
  background: url(fondo3.png) bottom right no-repeat,  
             url(fondo2.png) center no-repeat,  
             url(fondo1.gif) center repeat;  
  width: 300px;  
}  
</style>
```

Cabe señalar que las distintas imágenes de fondo se tienen que escribir en la declaración CSS separadas por comas. Además, las imágenes que declaramos se van colocando de modo que la primera aparece sobre las siguientes. Así pues, en esta declaración, fondo1.gif, que está colocada como último fondo, es la que aparece atrás de todo.



BORDES CON IMÁGENES

Border-image es un atributo que nos ayudará a aplicar nuevos estilos a las cajas con CSS, a través de la utilización de imágenes en los bordes de los elementos, incluso pudiendo escoger varias imágenes para varias de las partes de los bordes.

Colocar imágenes en los bordes es una tarea que ya se realiza en el diseño web, y para ello se suelen utilizar complementariamente técnicas con los lenguajes HTML y CSS.

Con CSS 3 vamos a poder hacer eso mismo simplemente escribiendo algunos nuevos atributos a los elementos que deseemos.

Distintas especificaciones sobre border-image

Las especificaciones de CSS 3 están en etapa de desarrollo. El organismo W3C, que se encarga de definir los estándares de Hojas de Estilo en Cascada, ha alterado algunas veces las especificaciones del atributo border-image y relacionados.

Es por ello que todavía hay algunas diferencias entre lo que los navegadores entienden con este atributo y lo que recomienda el W3C. Esto quiere decir que en el futuro todavía puede cambiar el funcionamiento de este atributo en los distintos navegadores, tal como anuncia Mozilla en su centro de desarrollo.

Las propiedades estándar tiene los siguientes atributos:

- **border-image:** Propiedad de borde general o completo.
- **border-top-image:** Propiedad de borde superior.
- **border-right-image:** Propiedad de borde derecho.
- **border-bottom-image:** Propiedad de borde inferior.
- **border-left-image:** Propiedad de borde izquierdo.
- **border-corner-image:** Propiedad de borde en esquinas.
- **border-top-left-image:** Propiedad de borde en esquina superior izquierda.
- **border-top-right-image:** Propiedad de borde esquina superior derecha.



- **border-bottom-left-image:** Propiedad de borde esquina inferior izquierda.
- **border-bottom-right-image:** Propiedad de borde esquina inferior derecha.

Pero como ya vimos, aun los navegadores no se apegaban a los estándares, por lo tanto para poder visualizar esta propiedad correctamente en la mayoría de los navegadores, debemos implementarla así:

Para Opera: **-o-border-image:**

Para Firefox: **-moz-border-image:**

Para Chrome/Safari: **-webkit-border-image:**

Por ejemplo, tendríamos este elemento HTML:

```
<div class="borde_imagen">Caja con borde con imágenes</div>
```

Y ahora podríamos aplicar estilos para crear un borde en la imagen:

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

```
.borde_imagen {
```

```
border-image: url("sello.png") 20 20 20 20 round round;  
-o-border-image: url("sello.png") 20 20 20 20 round round;  
-moz-border-image: url("sello.png") 20 20 20 20 round round;  
-webkit-border-image: url("sello.png") 20 20 20 20 round round;  
border-width: 2em;  
width: 600 px;
```



```
height: 300px;  
}
```

</style>

La imagen que estamos utilizando como borde es la siguiente:



Los valores de **20 20 20 20** indican el tamaño que se va a utilizar de cada esquina de la imagen, el orden va como las agujas del reloj, es decir, arriba, derecha, abajo e izquierda.

Luego donde dice **round** se determina cómo se van a escalar y cortar los bordes de los lados y superior e inferior de nuestro elemento. Hay tres posibles valores:

- **stretch** (estirar)
- **repeat** (repetir)
- **round** (redondear)

Esta especificado 2 veces porque uno corresponde a los lados y otro a las partes superior e inferior. Si especificamos sólo un valor es como poner dos veces el mismo (por ejemplo sólo una vez **round**. Entonces será **round** tanto para los lados superior e inferior como para los lados izquierda y derecha). El valor por defecto es “**stretch**”.



SOMBRAS

Hasta ahora aplicar un efecto de sombra a cualquier elemento de nuestro html era un proceso en donde teníamos que cargar imágenes creadas previamente en nuestro programa de edición de imágenes favorito, recortarlas, ajustarlas, etc.

Con el atributo *box-shadow* podemos aplicar sombras a nuestras cajas con facilidad. La sintaxis de *box-shadow* es la siguiente:

- `box-shadow: distanciaX distanciaY difuminado color;`

Desplazamiento horizontal de la sombra (distancia x): La sombra de un elemento suele estar un poco desplazada con respecto al elemento que la produce y su posición será en función del ángulo con el que llegue la luz. Si la sombra quisiéramos que apareciera un poco hacia la izquierda del elemento original que la produce, pondríamos un valor negativo a este atributo. Cuanto más desplazamiento tenga una sombra, el elemento que la produce parecerá que está más separado del lienzo de la página.

Desplazamiento vertical de la sombra (desplazamiento y): El segundo valor que colocamos en el atributo *box-shadow* es el desplazamiento vertical de la sombra con respecto a la posición del elemento que la produce. Este valor es similar al desplazamiento horizontal. Valores positivos indican que la sombra aparecerá hacia abajo del elemento y valores negativos harán que la sombra aparezca desplazada un poco hacia arriba.

Difuminado: El tercer valor indica cuánto queremos que esté difuminado el borde de la sombra. Si el difuminado fuera cero, querría decir que la sombra no tiene ningún difuminado y aparece totalmente definida. Si el valor es mayor que cero, quiere decir que la sombra tendrá un difuminado de esa anchura.

Color de la sombra: El último atributo que se indica en el atributo *box-shadow* es el color de la sombra. Generalmente las sombras en el mundo real tienen un color negro o grisáceo, pero con CSS3 podremos indicar cualquier gama de color para hacer la sombra, lo que nos dará bastante más



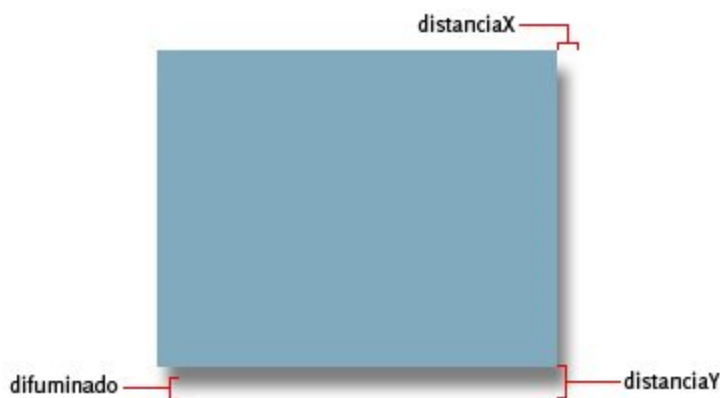
versatilidad a los diseños gracias a la posible utilización de sombras en distintos colores, que puedan combinar mejor con nuestra paleta. La sombra podemos indicarla en hexadecimal o en RGBA tal como vimos anteriormente para el color.

Aplicado de la siguiente forma:

Código:

```
#sombra {  
    box-shadow: 5px 10px 7px rgba(0,0,0,0.5);  
}
```

El resultado que obtendremos de la aplicación de ese estilo será algo más o menos así:



Si quisiéramos añadir varias sombras, sólo tendríamos que añadir un nuevo valor al atributo usando una coma como separador:

```
#sombra {  
    box-shadow: 5px 10px 7px rgba(0,0,0,0.5), -5px -10px 7px rgba(0,0,0,0.5);
```



}

Y el resultado que obtendríamos sería algo así:



Compatibilidad de las sombras CSS con navegadores

Por el momento podemos utilizar box-shadow en las versiones más modernas del navegador Opera. Por su parte, navegadores basados en Mozilla y WebKit tienen soporte a esta funcionalidad de CSS3, pero a través de unos atributos CSS con una ligera variación en su nombre.

Atributo box-shadow para navegadores basados en Mozilla, como Firefox: De manera temporal, Firefox es capaz de interpretar el atributo -moz-box-shadow, por ejemplo:

```
-moz-box-shadow: 1px 1px 0px #090;
```

Atributo box-shadow para navegadores basados en WebKit, como Safari o Google Chrome: En estos momentos y de manera temporal, navegadores como Chrome o Safari entienden el atributo: -webkit-box-shadow, por ejemplo:

```
-webkit-box-shadow: 3px 3px 1px #fc8;
```

Como podremos imaginar, si deseamos ampliar al máximo la compatibilidad con box-shadow, necesitaríamos indicar tanto el propio atributo box-shadow (que funciona en Opera y en el futuro funcionará en todos los navegadores), así como -moz-box-shadow y -webkit-box-shadow para que funcione en las versiones actuales de Firefox, Safari, Chrome, etc.



Brillo (resplandor exterior)

Si queremos crear un resplandor en vez de una sombra, tenemos que darle valor 0 a ambas distancias y aplicar un color claro.

Así que, haciendo uso de la misma propiedad CSS para la creación de sombras, podemos crear los resplandores que deseemos. Para apreciarlos mejor podemos colocar colores oscuros de fondo.

Estas son las propiedades CSS que necesitaremos para crear una sombra:

```
-moz-box-shadow: 0px 0px 30px #ffffff;  
-webkit-box-shadow: 0px 0px 30px #ffffff;  
box-shadow: 0px 0px 30px #ffffff;
```

Si queremos hacer el resplandor con otro color, por ejemplo verde, tendríamos que cambiar únicamente el RGB de las propiedades CSS, tal como se puede ver a continuación.

```
-moz-box-shadow: 0px 0px 30px #A3FF0F;  
-webkit-box-shadow: 0px 0px 30px #A3FF0F;  
box-shadow: 0px 0px 30px #A3FF0F;
```

Estas propiedades se verán en pantalla de esta forma:

Esta capa tiene un
resplandor exterior,
pero ahora he
utilizado un
resplandor de color
verde!!



FONDOS INTELIGENTES

CSS permite especificar que los elementos tengan un fondo con una imagen y además, con algunos atributos como `background-position` o `background-repeat` (atributos que ya trabajábamos desde CSS2) podemos definir cosas como la posición de la imagen de fondo se traslade a otro lugar o que se forme un mosaico. Esas propiedades son bastante utilizadas en el diseño de páginas web y quizás ya las dominemos. Ahora bien, con CSS 3 tenemos la posibilidad de especificar nuevos estilos o modos de comportamiento de las imágenes.

Exploraremos el nuevo atributo `background-origin`, que forma parte de la nueva especificación de CSS.

Background-origin

Cuando colocamos una imagen de fondo en un elemento de HTML se posiciona dentro del espacio de ese elemento. Por defecto una imagen de fondo aparece como un mosaico, repitiendo la imagen a lo largo de todo el espacio de ese elemento. Pero para colocar esa imagen el navegador utiliza un origen de coordenadas, que veríamos más fácilmente si hacemos que la imagen no se repita, es decir, que no se haga un mosaico en el fondo.

Con `background-origin` podemos definir el origen de coordenadas sobre el que se va a colocar la imagen de fondo, para que sea el borde del elemento, su padding o su contenido. Veamos sus posibles valores con una explicación:

border-box: Este valor marca que el origen de coordenadas de la imagen será el borde del elemento. En este caso estamos indicando que la imagen empiece donde empieza el borde del elemento, si es que tiene borde.

- `background-origin: border-box;`

padding-box: Este valor es el utilizado por defecto en CSS 3 y es tal como se posiciona la imagen en navegadores que sólo entienden CSS 2 o inferior. Cuando indicamos `background-origin: padding;`



queremos que el eje de coordenadas donde se va a colocar la imagen sea el espacio destinado al elemento, incluyendo su posible padding y sin contar el posible borde.

- background-origin: padding-box;

content-box: El tercero de los posibles valores de background-origin sirve para que el origen de coordenadas para la posición de la imagen de fondo sea el lugar donde empieza el contenido del elemento, es decir, sin tener en cuenta sus posibles bordes y padding.

- background-origin: content-box;

En Mozilla Firefox debemos utilizar el atributo -moz-background-origin, en navegadores basados en webkit (como Chrome o Safari) se debe utilizar el atributo **-webkit-background-origin**.

Además, en navegadores basados en webkit y Mozilla no tienen en cuenta la terminación "-box" al final de los valores de los atributos, por lo que hay que eliminarla.

Por su parte, en Opera ya tienen implementada la funcionalidad con el nombre de atributo definitivo, y sus valores con la correspondiente terminación "-box", que es como aparece en la especificación actual del modelo de caja de CSS 3.

En los siguientes ejemplos veremos el código CSS para aplicar distintos valores de background-origin y que los entiendan todos los navegadores más modernos.

Background-size

CSS3 permite especificar un tamaño a las imágenes de fondo. Este tamaño puede ser especificado en pixeles, (height y width), o en porcentaje. Si se especifica un tamaño en porcentaje, el tamaño es relativo al ancho o altura de la zona a la que se ha atribuido la propiedad background-origin.



Resumen

En esta Unidad...

En la presente unidad desarrollamos los conceptos necesarios para incorporar los atributos gráficos a nuestras estructuras de HTML utilizando el lenguaje CSS3

Con las propiedades propuestas podemos comenzar a plantear la estructura gráfica de una página web, de acuerdo a las nuevas normativas.

En la próxima Unidad...

En la próxima unidad vamos seguir incorporando nuevas propiedades de CSS3, particularmente la posibilidad de generar sitios multiplataforma (para visualizar en cualquier dispositivo).