

El canvas de HTML 5 (VI). Gradientes.



En el artículo anterior hemos aprendido a crear figuras rellenas, con un color sólido. En este artículo vamos a empezar a aprender como crear y usar gradientes, tema que completaremos en el próximo artículo.

Los gradientes, o degradados, como se les llama en algunos textos, son muy llamativos a la hora de crear figuras. Algunos diseñadores gráficos los detestan y, honestamente, un gradiente debe ser suave y estar bien concebido para causar una buena impresión visual. En eso, como en todo (y nunca mejor dicho), para gustos los colores. Aquí aprenderemos a crearlos y usarlos, y luego será tu talento artístico y tu experiencia los que te dirán cómo y cuándo emplearlos.

QUÉ SON LOS GRADIENTES

Aunque todos sabemos, desde el punto de vista del color, que son los gradientes, vamos a ver lo qué son desde el punto de vista del canvas de HTML 5. Sabemos que nosotros no trabajamos directamente sobre el lienzo, sino sobre un objeto de contexto. Bien, pues, a partir de este objeto se crean nuevos objetos que representan a los gradientes que vamos a emplear en nuestro diseño. Esta estructura de objetos que forman parte de otros objetos puede parecer un poco rebuscada pero, si has programado en Java (por ejemplo) esto es un paseo por el parque.

Existen dos tipos de gradientes: los **lineales**, en los que se empieza en un color y se va cambiando, de forma progresiva a otro, bien de lado a lado, o en diagonal, y los **radiales**, en los que el degradado tiene una forma circular, con un color en el centro otro color en el perímetro, y un paso, más o menos suave de uno a otro.

GRADIENTES LINEALES

Estos son los que vamos a conocer en este artículo. En primer lugar, veamos cómo se crean. Tal como hemos dicho, el gradiente es un objeto, que se crea dentro del objeto contexto, y que luego se aplica para rellenar una figura. Para crearlo seguimos la siguiente estructura general:

```
var gradiente_lineal = contexto_canvas.createLinearGradient(origen_x, origen_y, destino_x, ángulo);
gradiente_lineal.addColorStop(0, color_1);
gradiente_lineal.addColorStop(1, color_2);
```

Cómo ves, el objeto gradiente se crea con el método `contexto.createLinearGradient()`. Este es un poco complicado de entender al principio aunque, a base de experimentar, en seguida se comprende. El gradiente es un patrón, que incluye una anchura y un ángulo, de modo que podremos crearlo con el ancho de la figura que luego queramos rellenar con él. Si el ancho es diferente, el patrón se adaptará para rellenar adecuadamente la figura. El método para crear un gradiente lineal recibe cuatro argumentos:

El origen de **X** y el origen de **Y**. Este punto marca el origen del patrón y, en su uso más simple, coincidirá con las coordenadas **X** e **Y** del objeto a rellenar, por lo que, en nuestras primeras pruebas, pondremos estos valores a **0**.

El ancho del patrón. Es el ancho en píxeles que tendrán el patrón.

El ángulo. Si es **0**, el patrón tendrá el degradado de colores en vertical. Si le damos un valor, estaremos estableciendo el degradado en diagonal. Si el parámetro anterior es **0** y este último tiene un valor, el degradado será horizontal, de arriba a abajo.

Dicho así, suena un poco críptico, pero, en este mismo artículo, veremos un ejemplo que nos lo dejará todo claro, no te preocupes demasiado ahora. Como te he dicho, al principio cuesta un poco "pillarlo" pero, en cuanto experimentas un poco, en seguida se entiende.

A continuación, usamos un método llamado `addColorStop()`, que pertenece al objeto gradiente que acabamos de crear, para añadir cada color que queramos al degradado. Como mínimo, por supuesto, debe haber dos colores. No hay límite máximo, pero un degradado con un exceso de tonos, podría quedar hasta desagradable a la vista. Este método recibe dos argumentos:

El primero indica donde se inicia el color al que nos referimos, con respecto al gradiente. Este es un número entre 0 y 1, que se refiere a la parte porcentual con respecto al tamaño que le hemos dado al gradiente. Así, por ejemplo, si cuando creamos el objeto de gradiente establecimos, como tercer parámetro, 200, y el primer parámetro de `addColorStop()` es 0.3, nos estamos refiriendo al 30% de 200, es decir, a 60 píxeles (en realidad, sólo son píxeles, si la figura a rellenar tiene el mismo ancho, si no, es un valor relativo al ancho de la figura).

El segundo parámetro es el color que queremos añadir al gradiente.

ACLARANDO CONCEPTOS

Acabamos de soltar una parrafada teórica que, si bien al finalizar este artículo, entenderemos perfectamente, de momento a la mayoría le sonará a nada parecido a nada. Así que vamos a dejarnos de historias, y vamos a ver un ejemplo, llamado [canvas_010.htm](#):

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<title>Canvas</title>
```

```
<link rel="stylesheet" href="css/estilos_010.css">
```

```
</head>
```

```
<body>
```

```
<canvas id="canvas_1" height="300">
```

Tu navegador no soporta características esenciales de HTML 5.

```
</canvas>
```

```
<script src="https://code.jquery.com/jquery-3.2.1.min.js"></script>
```

```
<script language='javascript'>
```

```
/* Referenciamos el canvas y, si se han referenciado correctamente, creamos el correspondiente contexto.
```

```
Si el navegador soporta canvas, la referenciación no presenta problemas. */
```

```
var objeto_canvas_1 = $('#canvas_1');
```

```
if ($(objeto_canvas_1)[0].getContext) var contexto_canvas_1 = $(objeto_canvas_1)[0].getContext('2d');
```

```
/* Definimos un gradiente lineal. Estos se crean sobre la base de un tamaño determinado, aunque luego puedan adaptarse a otros tamaños al rellenar con ellos una figura.
```

```
Los gradientes son objetos que, a su vez, forman parte del objeto contexto de cada canvas.
```

```
Aquí vamos a crear tres gradientes: uno recto y dos en ángulo.
```

```
Tendrán dos colores cada uno, y el tercero cuatro colores.*/
```

```
// El primero, recto de azul a gris.
```

```
var gradiente_lineal_recto = contexto_canvas_1.createLinearGradient(0, 0, 280, 0);
```

```
gradiente_lineal_recto.addColorStop(0, "#5400FF"); // Azul
```

```
gradiente_lineal_recto.addColorStop(1, "#B2B2B2"); // Gris
```

```
// El segundo en ángulo de rojo a amarillo.
```

```
var gradiente_lineal_angulo_1 = contexto_canvas_1.createLinearGradient(0, 0, 280, 80);
```

```
gradiente_lineal_angulo_1.addColorStop(0, "#FF0000"); // Rojo
```

```
gradiente_lineal_angulo_1.addColorStop(1, "#FFF600"); // Amarillo
```

```
// El tercero en ángulo con cinco colores
```

```
var gradiente_lineal_angulo_2 = contexto_canvas_1.createLinearGradient(10, 210, 280, 80);  
gradiente_lineal_angulo_2.addColorStop(0.30, "#FF0000"); // Rojo  
gradiente_lineal_angulo_2.addColorStop(0.35, "#FFF600"); // Amarillo  
gradiente_lineal_angulo_2.addColorStop(0.5, "#22B106"); // Verde  
gradiente_lineal_angulo_2.addColorStop(0.75, "#F80AF5"); // Morado  
gradiente_lineal_angulo_2.addColorStop(1, "#5400FF"); // Azul
```

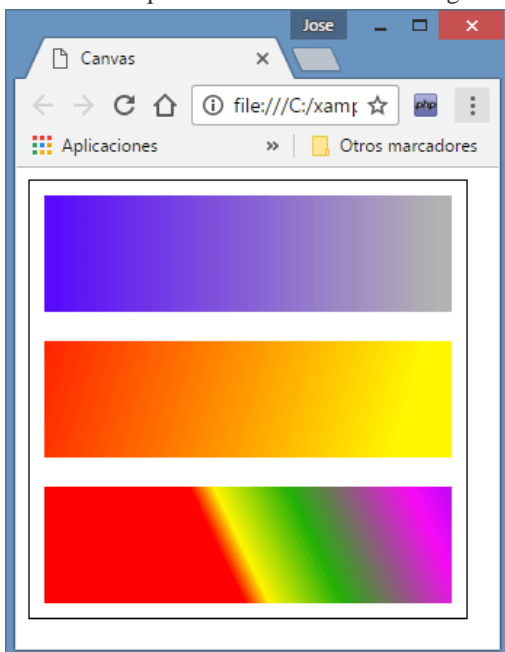
```
// Una figura con el primer gradiente  
contexto_canvas_1.beginPath();  
contexto_canvas_1.fillStyle = gradiente_lineal_recto;  
contexto_canvas_1.rect(10, 10, 280, 80);  
contexto_canvas_1.fill();
```

```
// Una figura con el segundo gradiente  
contexto_canvas_1.beginPath();  
contexto_canvas_1.fillStyle = gradiente_lineal_angulo_1;  
contexto_canvas_1.rect(10, 110, 280, 80);  
contexto_canvas_1.fill();
```

```
// Una figura con el tercer gradiente  
contexto_canvas_1.beginPath();  
contexto_canvas_1.fillStyle = gradiente_lineal_angulo_2;  
contexto_canvas_1.rect(10, 210, 280, 80);  
contexto_canvas_1.fill();
```

```
</script>  
</body>  
</html>
```

El resultado que obtenemos con este código es como el que ves a continuación:



Empieza a mirar el código. Tenemos un sólo canvas y un solo objeto contexto pero, a partir de él, hemos creado tres objetos de

gradiente diferentes, como ves entre las líneas 30 y 45.

A partir de ahí, para usar cada gradiente, solo tenemos que aplicarlo como valor de la propiedad `fillStyle`, en cada caso, como puedes ver que hacemos en las líneas resaltadas. Una vez establecido este valor, el gradiente indicado se empleará para rellenar cada figura. Y ahora es donde tú tienes que poner un poco de esfuerzo de tu parte, para entender la teoría que comentábamos en el apartado anterior. Te toca ir a donde se definen los gradientes y los colores empleados, y empezar a cambiar valores, actualizando tu página, para ver como reacciona a los cambios. No te preocupes si el resultado es un desastre al principio. Siempre puedes partir del código inicial. Lo que importa es que cambies las anchuras y alturas y los puntos de origen en la creación de cada objeto gradiente, así como los puntos donde se añaden los colores, y veas el resultado en cada caso. Créeme. Es la única manera real de entender como funciona esto y, además, la más rápida y eficiente.

[En este enlace](#) te he dejado el código original para descargar. En el próximo artículo hablaremos de degradados radiales, y veremos como aplicar degradados a los contornos (`stroke`), y no solo a los rellenos.