



APRENDERAPROGRAMAR.COM

API CANVAS HTML Y
JAVASCRIPT. EJEMPLOS.
DIBUJAR CÍRCULOS,
FORMAS, GRÁFICOS,
ANIMACIONES, JUEGOS,
ETC. (CU01195E)

Sección: Cursos

Categoría: Tutorial básico del programador web: JavaScript desde cero

Fecha revisión: 2029

Resumen: Entrega nº95 del Tutorial básico “JavaScript desde cero”.

Autor: César Krall

CANVAS COMO SUPERFICIE GRÁFICA

En el pasado, la creación de efectos gráficos y animaciones no resultaba sencilla dentro de páginas web. Un recurso que fue muy utilizado fue insertar applets Java u objetos o películas flash (archivos swf), con el inconveniente de que esto implicaba hacer uso de Flash Player, un plugin adicional que se debía instalar en los navegadores. La tendencia actual es que la animación gráfica quede dentro de los estándares web definidos por HTML, JavaScript y CSS sin dependencias de terceras partes.



Dentro del desarrollo del estándar de HTML se ha incluido una nueva etiqueta o elemento HTML: `<canvas>`. Esta etiqueta funciona igual que cualquier otra en el sentido de que se abre como `<canvas>` y se cierra como `</canvas>`.

El elemento canvas es un lienzo encima del cual se pueden dibujar gráficos y animaciones. Permite generar gráficos en pantalla, crear animaciones, manipular imágenes e incluso video, etc. En este sentido, permite generar las animaciones gráficas necesarias utilizando HTML+JavaScript+CSS, de modo que todos los desarrollos web puedan completarse con lo que se denominan lenguajes web sin necesidad de recurrir a tecnologías de terceras partes como Flash o los applets de Java.

La incorporación de toda esta potencia gráfica corre a cargo de los navegadores. Dada la complejidad que conlleva, todo lo relacionado con canvas se dice que constituye una API (interfaz de programación de aplicaciones). Algunos navegadores antiguos no admiten los elementos `<canvas>`, pero todos los navegadores modernos sí lo hacen.

El elemento canvas constituye un soporte al que se le da uso a través de JavaScript.

Nosotros en este curso no vamos a estudiar la api Canvas. Simplemente nos limitaremos a citar algunas de las posibilidades que el uso de este elemento junto a JavaScript permite para los desarrollos web. La creación de elementos gráficos avanzados y animaciones avanzadas constituye un área de especialización dentro de los desarrollos web, por tanto no lo consideramos un conocimiento "básico". No obstante, es adecuado tener una idea de qué se puede hacer con canvas y JavaScript.

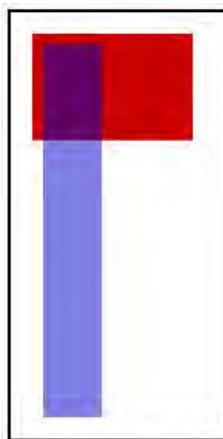
UTILIZANDO CANVAS

Un elemento HTML canvas tiene propiedades y métodos que podemos usar a través de JavaScript, de la misma forma que usamos propiedades y métodos sobre el resto de elementos HTML como ``, `<input>`, `<label>`, etc.

No vamos a detenernos a estudiar el API Canvas, pero queremos mostrar brevemente algunos ejemplos que nos den una idea de qué es y para qué sirve. Escribe este código en un editor de textos y guárdalo con un nombre de archivo como ejemplo1.html:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8"/>
<script type="text/javascript">
function dibujar() {
var canvas = document.getElementById("canvas");
var ctx = canvas.getContext("2d");
ctx.fillStyle = "rgb(200,0,0)";
ctx.fillRect (20, 20, 150, 100);
ctx.fillStyle = "rgba(0, 0, 200, 0.5)";
ctx.fillRect (30, 30, 55, 350);
}
</script>
</head>
<body onload="dibujar();">
<canvas id="canvas" width="200" height="400" style="border-style:solid;">
Su navegador no tiene soporte para mostrar el contenido
</canvas>
</body>
</html>
```

El resultado obtenido será algo similar a esto:



Si investigas un poco sobre el código podrás comprender con facilidad qué es lo que hace. No vamos a entrar en detalles porque no es el objetivo de este curso.

Quizás pienses que esto es relativamente poco útil, sin embargo hay un gran potencial detrás del elemento canvas unido a la potencia de JavaScript. La creación anterior implica que generamos una imagen a partir de código, esto implica un potencial enorme, ya que la imagen puede estar ligada a datos (introducidos por el usuario, extraídos de una base de datos, tomados de otra página web, etc.), y también ligada a eventos (lo cual permite crear movimientos en respuesta a la acción del usuario, incluso crear juegos) o ligada al tiempo (animaciones, imágenes que van cambiando en el tiempo).

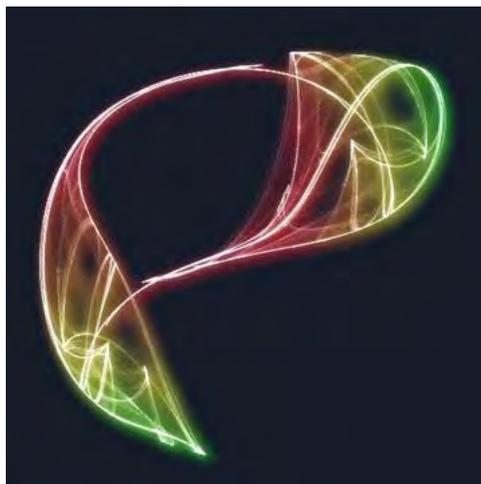
Escribe este código en un editor de textos y guárdalo con un nombre de archivo como ejemplo2.html:

```
<html>
<head><title>Ejemplo canvas de w3.org curso JavaScript aprenderaprogramar.com </title></head>
<body>
<h1> Ejemplo canvas curso JavaScript aprenderaprogramar.com </h1>
<canvas id="canvasGlowing" width="800" height="450">
El navegador no admite canvas</canvas>
</body>
<script>
var canvas = document.getElementById('canvasGlowing');
var context = canvas.getContext('2d');
var lastX = context.canvas.width * Math.random();
var lastY = context.canvas.height * Math.random();
var hue = 0;
function line() {
  context.save();
  context.translate(context.canvas.width/2, context.canvas.height/2);
  context.scale(0.9, 0.9);
  context.translate(-context.canvas.width/2, -context.canvas.height/2);
  context.beginPath();
  context.lineWidth = 5 + Math.random() * 10;
  context.moveTo(lastX, lastY);
  lastX = context.canvas.width * Math.random();
  lastY = context.canvas.height * Math.random();
  context.bezierCurveTo(context.canvas.width * Math.random(),
    context.canvas.height * Math.random(),
    context.canvas.width * Math.random(),
    context.canvas.height * Math.random(),
    lastX, lastY);

  hue = hue + 10 * Math.random();
  context.strokeStyle = 'hsl(' + hue + ', 50%, 50%)';
  context.shadowColor = 'white';
  context.shadowBlur = 10;
  context.stroke();
  context.restore();
}
setInterval(line, 150);

function blank() {
  context.fillStyle = 'rgba(0,0,0,0.1)';
  context.fillRect(0, 0, context.canvas.width, context.canvas.height);
}
setInterval(blank, 140);
</script>
</html>
```

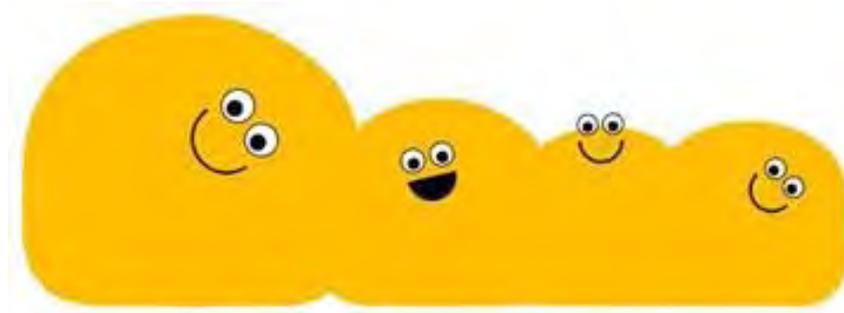
El resultado esperado es que se dibujen líneas creando un efecto tridimensional en la pantalla (ten en cuenta que en navegadores antiguos no se mostrará el resultado, sólo los navegadores modernos admiten canvas).



¿PARA QUÉ SIRVE CANVAS?

El api Canvas combinado con CSS, HTML y JavaScript nos permite hacer todo aquello que se nos pueda ocurrir. Vamos a citar algunas de las aplicaciones habituales:

- a) Dibujar formas (líneas, puntos, círculos, etc.)
- b) Dibujar texto
- c) Dibujar encima de imágenes o manipular imágenes
- d) Dibujar gráficos
- e) Crear animaciones de todo tipo (matemáticas, escolares, publicidad, etc.).
- f) Crear espacios tridimensionales por donde el usuario puede moverse
- g) Crear juegos
- h) Manipular videos por ejemplo crear efectos gráficos sobre ellos
- i) Hacer todo lo anterior en respuesta a eventos (por interacción con el usuario)
- j) Hacer todo lo anterior ligado a datos variables



Las bolas de la imagen anterior se pueden dibujar con relativa facilidad con Canvas. También se pueden dotar de animación, hacer que salten, se dividan, etc.

Si escribes canvas examples en un buscador como google o bing podrás acceder a numerosos ejemplos de uso de canvas. Muchos de ellos facilitan el código fuente de forma ordenada para que te sea fácil reproducirlo en tu ordenador o incorporarlo a tus páginas web.

VENTAJAS E INCONVENIENTES DE CANVAS

La principal ventaja de Canvas es permitir la creación y manipulación de todo tipo de formas e imágenes sin dependencias de plugins o tecnologías externas a los propios lenguajes de desarrollo web y navegadores. Igualmente es una gran ventaja la facilidad que brinda para la interacción gráfica con el usuario o para ligar el aspecto gráfico a datos que cambian dinámicamente.

Canvas es una herramienta que, al igual que todo, bien usado resulta muy útil y mal usado puede generarnos problemas. Un inconveniente ligado a canvas es pretender hacer páginas webs muy animadas y con esto sobrecargar el computador del usuario. Ten en cuenta que crear animaciones consume recursos y puede ralentizar la navegación por las páginas web o crear molestias al usuario.

EJERCICIO OPCIONAL

Este ejercicio no forma parte del curso, por eso se califica como opcional. El ejercicio consiste en dibujar la bandera de tu país usando canvas y JavaScript. La respuesta al ejercicio debe ser el código que genere la bandera.

Para comprobar si tus respuestas y código son correctos puedes consultar en los foros aprenderaprogramar.com.

Próxima entrega: CU01196E

Acceso al curso completo en aprenderaprogramar.com -- > Cursos, o en la dirección siguiente:
http://aprenderaprogramar.com/index.php?option=com_content&view=category&id=78&Itemid=206