CANVAS

Autor: Camila Andrea Amaya Vélez

*Ingeniería de Sistemas y Computación, Universidad Tecnológica de Pereira, Pereira, Colombia*

*Correo-e: c.amaya@utp.edu.co*

### INTRODUCCIÓN

Canvas es un elemento muy importante de HTML que permite el diseño de gráficos y animaciones de forma dinámica, también sirve a los usuarios de canvas para hacer collage de información que han obtenido de diferentes fuentes el usuario a través de canvas puede empalmar imágenes textos y/o documentos.

### HISTORIA DE CANVAS

Canvas fue introducido inicialmente introducido por Apple para su uso dentro de su propio componente mac OS X, para empujar aplicaciones como widgest, de Dashboard y el navegador safari, luego se adoptó en la versión 1.8 de los navegadores Gecko y Opera. Fue estandarizado por el grupo de trabajo de tecnologías de aplicación de hipertexto web sobre las nuevas especificaciones propuestas para tecnologías web de última generación.

1. ¿QUE ES CANVAS?

Canvas es un elemento HTML incorporado en HTML5 que permite la generación de gráficos dinámicamente por scripting, entre otras cosas permite la renderización interpretada dinámica de gráficos 2D y mapas de bits. Así como animaciones con estos gráficos, se trata de un procedimiento de bajo nivel, que actualiza un mapa de bits y no tiene una gráfica de escena integrada.

Cualquier cosa que se escriba entre la apertura y el cierre de la etiqueta canvas solamente será interpretado con navegadores que no soportan aun la nueva etiqueta.

Cuando se trabaja con canvas realmente todo el trabajo se hace en JavaScript, primero se debe referenciar el elemento canvas y adquirir su contexto.



1. DESCRIPCIÓN

* Un elemento canvas es parecido al elemento <img> pero la diferencia es, que este no usa src y alt; este modelo canvas usa solo dos atributos –width y –height. Ambos son opcionales y pueden ser definidos usando propiedades DOM.
* Cuando los atributos alto y ancho no están definidos, el lienzo se iniciara en 300 pixeles ancho y 150 pixeles de alto.
* El elemento puede ser redimensionado por CSS pero, si el tamaño del CSS no respeta las medidas del canvas inicial, este aparecerá distorsionado.
* El atributo id no está especificado para el elemento pero es uno de los atributos globales de HTML; siempre es bueno proporcionar un id porque esto hace más fácil la identificación en un Script.
* El elemento canvas puede ser utilizado como a cualquier imagen normal.
* Se diferencia de un tag en, como los elementos videos, audio e imagen, es fácil definir contenido alternativo para mostrarse en navegadores viejos que no soporten el elemento canvas. Proporcionar contenido alternativo, es muy explícito; solo debemos insertar el contenido alterno dentro del elemento canvas. Los elementos que no soporten el elemento canvas ignoraran el contenedor y mostraran el contenido indicado dentro de este.

<canvas id="stockGraph" width="150" height="150">

Current stock price: $3.15 +0.15

</Canvas>

<canvas id="clock" width="150" height="150">

<img src="images/clock.png" width="150" height="150" alt=""/>

</canvas>

1. EJEMPLOS DE LIENZOS

Un lienzo es un área rectangular en una pagina HTML. Por defecto, un lienzo no tiene bordes ni contenido.

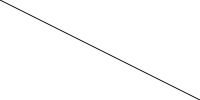
<canvas id="myCanvas" width="200" height="100"></canvas>

Siempre especifique el id atributo (al que se hará referencia en un script), y un atributo width y height para definir el tamaño del lienzo. Para agregar un borde, use el style atributo.

<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">  
</canvas>

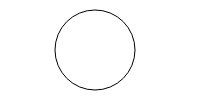
PARA DIBUJAR UNA LINEA.

var c=document.getElementById("myCanvas");  
var ctx=c.getContext("2d");  
ctx.moveTo(0, 0);  
ctx.lineTo(200, 100);  
ctx.stroke();



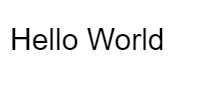
PARA DIBUJAR UN CIRCULO.

var c = document.getElementById("myCanvas");  
var ctx=c.getContext("2d");  
ctx.beginPath();  
ctx.arc(95, 50, 40, 0, 2 \* Math.PI);  
ctx.stroke();



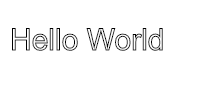
DIBUJAR UN TEXTO.

var c=document.getElementById("myCanvas");  
var ctx=c.getContext("2d");  
ctx.font = "30px Arial";  
ctx.fillText("Hello World", 10, 50);



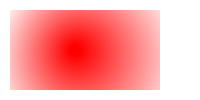
TEXTO DE TRAZO.

var c=document.getElementById("myCanvas");  
var ctx=c.getContext("2d");  
ctx.font = "30px Arial";  
ctx.strokeText("Hello World", 10, 50);



DIBUJAR DEGRADADO LINEAL.

var c = document.getElementById("myCanvas");  
var ctx = c.getContext("2d");  
  
// Create gradient  
var grd = ctx.createRadialGradient(75, 50, 5, 90, 60, 100);  
grd.addColorStop(0, "red");  
grd.addColorStop(1, "white");  
  
// Fill with gradient  
ctx.fillStyle = grd;  
ctx.fillRect(10, 10, 150, 80);



HTML

<canvas id="canvas"></canvas>

JAVASCRIPT

var canvas = document.getElementById('canvas');

var ctx = canvas.getContext('2d');

ctx.fillStyle = 'green';

ctx.fillRect(10, 10, 100, 100);

1. CSS

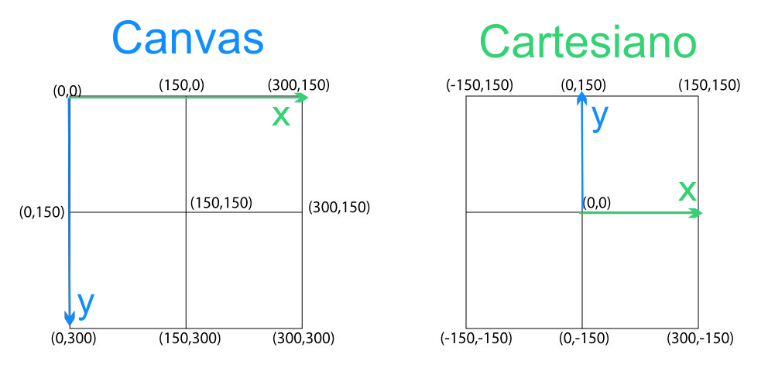
CSS es un lenguaje que describe el estilo de un documento HTML y describe como se deben mostrar los elementos HTML.

1. CONTEXTOS DE CANVAS

* 2d: guía para la creación de objetos representados en dos dimensiones.
* Webgl: con el cual puedes crear representaciones en tres dimensiones. Solo disponible a navegadores que soporten Webgl en su versión 1.
* Webgl2: con el cual puedes crear representaciones en tres dimensiones. Solo disponible a navegadores que soporten Webgl en su versión 2.
* Bitmaprenderer: con el cual puedes crear mapa de bits especiales para imágenes.

EJEMPLO 1:

En este ejemplo se utiliza el contexto 2d para la realización del ejercicio. Canvas es un elemento que crea un recuadro o lienzo en el cual se puede graficar, mediante el uso de coordenadas como las del plano cartesiano que alguna vez todos vimos, pero con la dirección del eje y invertida.



Mientas las coordenadas de un cartesiano se incrementan hacia arriba y a la derecha, en un lienzo canvas las coordenadas se incrementan hacia abajo y a la derecha.

canvas {

display: block;

margin: auto;

align: center;

}

body {

background-color: black;

}

var cnv = document.getElementById("plano");

var ctx = cnv.getContext('2d');

var img = new Image();

img.src = 'http://[i63.tinypic.com/rvm0lh.png](http://i63.tinypic.com/rvm0lh.png)';

img.onload = function() {ctx.drawImage(img,0,0);}

1. REFERENCIAS

[1] <https://www.w3schools.com/>

[2] <https://es.wikipedia.org/>

[3] <https://www.genbeta.com/>

[4] <https://developer.mozilla.org/>

[5]<https://devcode.la/tutoriales/animacion-con-canvas-de-html5/>