# 1. Canvas

## **What is Canvas?**

The HTML5 <canvas> element is used to draw graphics, on the fly, via scripting (usually JavaScript).

The <canvas> element is only a container for graphics. You must use a script to actually draw the graphics.

Canvas has several methods for drawing paths, boxes, circles, characters, and adding images.

**Create a Canvas**

A canvas is a rectangular area on an HTML page, and it is specified with the <canvas> element.

**Example 1-1**

<!DOCTYPE html>

<html>

<body>

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

Your browser does not support the HTML5 canvas tag.

</canvas>

</body>

</html>

**Draw Onto The Canvas With JavaScript**

**Example 1-2**

<!DOCTYPE html>

<html>

<body>

<canvas id="myCanvas" width="200" height="100" style="border:1px solid #c3c3c3;">

Your browser does not support the HTML5 canvas tag.

</canvas>

<script>

var c=document.getElementById("myCanvas");

var ctx=c.getContext("2d");

ctx.fillStyle="#FF0000";

ctx.fillRect(0,0,150,75);

</script>

</body>

</html>

**Example explained:**

First, find the <canvas> element:

var c=document.getElementById("myCanvas");

Then, call its getContext() method (you must pass the string "2d" to the getContext() method):

var ctx=c.getContext("2d");

The getContext("2d") object is a built-in HTML5 object, with many properties and methods for drawing paths, boxes, circles, text, images, and more.

The next two lines draw a red rectangle:

ctx.fillStyle="#FF0000";  
ctx.fillRect(0,0,150,75);

The fillStyle property can be a CSS color, a gradient, or a pattern. The default fillStyle is #000000 (black).

The fillRect(*x,y,width,height*) method draws a rectangle filled with the current fill style.

**Canvas Coordinates**

The canvas is a two-dimensional grid.

The upper-left corner of the canvas has coordinate (0,0)

So, the fillRect() method above had the parameters (0,0,150,75).

This means: Start at the upper-left corner (0,0) and draw a 150x75 pixels rectangle.

**Canvas - Paths**

To draw straight lines on a canvas, we will use the following two methods:

* moveTo(*x,y*) defines the starting point of the line
* lineTo(*x,y*) defines the ending point of the line

To actually draw the line, we must use one of the "ink" methods, like stroke().

We may use fill() to fill up the polygon produced by the points of the path.

## **Example 1-3**

Define a starting point in position (0,0), and an ending point in position (200,100). Then use the stroke() method to actually draw the line:

<!DOCTYPE html>

<html>

<body>

<canvas id="myCanvas" width="200" height="100" style="border:1px solid #d3d3d3;">

Your browser does not support the HTML5 canvas tag.</canvas>

<script>

var c=document.getElementById("myCanvas");

var ctx=c.getContext("2d");

ctx.moveTo(0,0);

ctx.lineTo(200,100);

ctx.stroke();

</script>

</body>

</html>

To draw a circle on a canvas, we will use the following method:

* arc(x,y,r,start,stop)

To actually draw the circle, we must use one of the "ink" methods, like stroke() or fill().

<!DOCTYPE html>

<html>

<body>

<canvas id="myCanvas" width="200" height="100" style="border:1px solid #d3d3d3;">

Your browser does not support the HTML5 canvas tag.</canvas>

<script>

var c=document.getElementById("myCanvas");

var ctx=c.getContext("2d");

ctx.beginPath();

ctx.arc(95,50,40,0,2\*Math.PI);

ctx.stroke();

ctx.fillStyle="red";

ctx.fill();

</script>

</body>

</html>

**Canvas - Text**

To draw text on a canvas, the most important property and methods are:

* font - defines the font properties for text
* fillText(*text,x,y*) - Draws "filled" text on the canvas
* strokeText(*text,x,y*) - Draws text on the canvas (no fill)

<!DOCTYPE html>

<html>

<body>

<canvas id="myCanvas" width="200" height="100" style="border:1px solid #d3d3d3;">

Your browser does not support the HTML5 canvas tag.</canvas>

<script>

var c=document.getElementById("myCanvas");

var ctx=c.getContext("2d");

ctx.font="30px Arial";

ctx.fillText("Hello World",10,50);

</script>

</body>

</html>

Using strokeText():

## **Example 1-4**

Write a 30px high text (no fill) on the canvas, using the font "Arial":

<!DOCTYPE html>

<html>

<body>

<canvas id="myCanvas" width="200" height="100" style="border:1px solid #d3d3d3;">

Your browser does not support the HTML5 canvas tag.</canvas>

<script>

var c=document.getElementById("myCanvas");

var ctx=c.getContext("2d");

ctx.font="30px Arial";

ctx.strokeText("Hello World",10,50);

</script>

</body>

</html>

**Canvas - Gradients**

Gradients can be used to fill rectangles, circles, lines, text, etc. Shapes on the canvas are not limited to solid colors.

There are two different types of gradients:

* createLinearGradient(*x,y,x1,y1*) - Creates a linear gradient
* createRadialGradient(*x,y,r,x1,y1,r1*) - Creates a radial/circular gradient

Once we have a gradient object, we must add two or more color stops.

The addColorStop() method specifies the color stops, and its position along the gradient. Gradient positions can be anywhere between 0 to 1.

To use the gradient, set the fillStyle or strokeStyle property to the gradient, and then draw the shape, like a rectangle, text, or a line.

Using createLinearGradient():

<!DOCTYPE html>

<html>

<body>

<canvas id="myCanvas" width="200" height="100" style="border:1px solid #d3d3d3;">

Your browser does not support the HTML5 canvas tag.</canvas>

<script>

var c=document.getElementById("myCanvas");

var ctx=c.getContext("2d");

// Create gradient

var grd=ctx.createLinearGradient(0,0,200,0);

grd.addColorStop(0,"red");

grd.addColorStop(1,"white");

// Fill with gradient

ctx.fillStyle=grd;

ctx.fillRect(10,10,150,80);

</script>

</body>

</html>

Using createRadialGradient():

## **Example 1-5**

Create a radial/circular gradient. Fill rectangle with the gradient:

<!DOCTYPE html>

<html>

<body>

<canvas id="myCanvas" width="200" height="100" style="border:1px solid #d3d3d3;">

Your browser does not support the HTML5 canvas tag.</canvas>

<script>

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);

</script>

</body>

</html>

**Canvas – Images**

## **Example 1-6**

To draw an image on a canvas, we will use the following method:

* drawImage(*image,x,y,width,length*)

<!DOCTYPE html>

<html>

<body>

<p>Image to use:</p>

<img id="NWLOGO" src="NWLOGO.jpg" alt="NW LOGO" width="300" height="200"><p>Canvas:</p>

<canvas id="myCanvas" width="350" height="300" style="border:1px solid #d3d3d3;">

Your browser does not support the HTML5 canvas tag.</canvas>

<script>

window.onload = function() {

var c=document.getElementById("myCanvas");

var ctx=c.getContext("2d");

var img=document.getElementById("NWLOGO");

ctx.drawImage(img,10,10,350,300);

}

</script>

</body>

</html>