HTML <canvas> Tag How To

#canvas# creates a 2-D drawing surface #canvas# description, syntax, usage, attributes and examples HTML-5.com is a great guide for web developers. TV Series & Actors and Actresses. Follow TV Series and HTML 5 on Google+.

HTML-5.com ➢ itemscopehttp://data-vocabulary.org/BreadcrumbHTML 5 ➢ itemscopehttp://data-vocabulary.org/BreadcrumbHTML Tags ➢ itemscopehttp://data-vocabulary.org/Breadcrumb **HTML <canvas> Tag How To**

➢ itemscopehttp://data-vocabulary.org/BreadcrumbExamples

The <canvas> Tag in HTML 5

The **canvas** element creates a two-dimensional drawing surface which can be used to draw images using 2-D drawing functions. It can be used to create images dynamically, which allows developers to:

- create charts and graphs online
- display graphics in online games
- display text as an image with various image effects such as drop shadow
- generate images on the fly

As a good example of the <canvas> tag, the site logo in the upper left corner of this page is text drawn on a canvas, not an image displayed with the <img> tag. Here is another example of the <canvas> tag.

```
<canvas id="ex1canvas" width="265" height="55" style="border: 1px
solid black"></canvas> <iframe src="about:blank" width="0" height="0"
onload="ctx=document.getElementById('ex1canvas').getContext('2d'); ctx.clearRect(0,
0, 265, 55); ctx.textBaseline = 'top'; ctx.shadowOffsetX = 6; ctx.shadowOffsetY
= 6; ctx.shadowBlur = 6; ctx.shadowColor = '#000000'; ctx.font = 'normal
36px sans-serif'; ctx.strokeStyle = '#006600'; ctx.strokeText('Hello', 12, 4);
ctx.font = 'bold 36px sans-serif'; var grad = ctx.createLinearGradient(100, 0,
220, 0); grad.addColorStop(0.0, '#000033'); grad.addColorStop(0.2, '#6699ff');
grad.addColorStop(0.4, '#003366'); grad.addColorStop(0.6, '#6699ff');
grad.addColorStop(1.0, '#000033'); ctx.fillStyle = grad; ctx.fillText('world!', 100, 4); grad
= ctx.createRadialGradient(227, 15, 5, 235, 25, 24); grad.addColorStop(0, 'rgba(198,
198, 255, 1.0)'); grad.addColorStop(0.9, 'rgba(0, 50, 255, 1.0)'); grad.addColorStop(1,
'rgba(0, 50, 255, 0)'); ctx.fillStyle=grad; ctx.fillRect(209, 0, 260, 50);"></iframe>
```

This is an actual working demo of the code <canvas> example code below. (Do *View Source* to verify that this page is using the HTML 5 DOCTYPE. You can also verify it is Valid HTML 5 using the HTML Validator. Try using it to validate URLs with HTML examples from other places that claim to be HTML 5 web sites!)

Another way to create a glow gradient on an object is shown in the SVG demo.

back to top

<canvas> Tag Syntax

Rules for coding HTML `canvas` elements

```
<a mode="pre" href="../body-tag/index.html#syntax"><body></a>
    <a mode="pre" href="../html-tag-list.html#flow-content-model">... flow content expected ...</a>
    <b mode="pre"><canvas width="<var>width</var>" height="<var>height</var>"></b>
       <a mode="pre" href="../../definitions/index.html#flow-content">... flow content ...</a>
    <b mode="pre"></canvas></b>
    ...
    <a mode="pre" href="../html-tag-list.html#phrasing-content-model">... phrasing content expected .
    <a mode="pre" href="../../definitions/index.html#phrasing-content">... phrasing content ...</a><b
    ...
  </body>
```

Rules for coding the HTML canvas element

Make sure you understand the [difference between a tag and element](#) and are familiar with the [definitions](#) of [namespace](#) and other [HTML terms](#).

1. Since the **canvas** [element](#) has a [transparent content model](#), it can be coded anywhere [phrasing content](#) is allowed, including where [flow content](#) is expected.
2. Begin the **canvas** [element](#) with a starting <canvas> tag. The element name uses lower case letters and should be in the [HTML namespace](#), which it will pick up automatically from the [xmlns attribute](#) on the [<html> tag](#).
3. Inside the starting `<canvas>` tag, code the desired width and height of the canvas in the [width and height attributes](#).
4. Inside the **canvas** [element](#), between the starting `<canvas>` tag and the ending `</canvas>` tag, code the fallback content. The fallback content should convey to the user information that is equivalent to what is displayed on the canvas.
5. End the **canvas** [element](#) with a matching `</canvas>` closing tag.

<canvas> Content Model

When coded in [flow content](#), the content of the **canvas** element can include [HTML comments](#), [text content](#) and any [HTML tags that can be used in flow content](#).

When coded in [phrasing content](#), the content of the **canvas** element can include [HTML comments](#), [text content](#) and only those [HTML tags that can be used in phrasing content](#).

[back to top](#)

<canvas> Tag Attributes

Attributes of the <canvas> tag

[global attributes](#)	In addition to the personal attributes of the <canvas> tag below, any of the [common HTML attributes](#) can also be coded.
`width`, `height`	

[back to top](#)

<canvas> Tag Examples

Examples of the `canvas` tag in HTML 5

Hello World! demo of <canvas> tag with gradients and drop shadow

```
<canvas id="ex1canvas" width="265" height="55" style="border: 1px solid black"/>
<iframe src="about:blank" width="0" height="0"
   onload="ctx=document.getElementById('ex1canvas').getContext('2d');
      ctx.clearRect(0, 0, 265, 55);
      ctx.textBaseline = 'top';
      ctx.shadowOffsetX = 6;
      ctx.shadowOffsetY = 6;
      ctx.shadowBlur = 6;
      ctx.shadowColor = '#000000';
      ctx.font = 'normal 36px sans-serif';
      ctx.strokeStyle = '#006600';
      ctx.strokeText('Hello', 12, 4);
      ctx.font = 'bold 36px sans-serif';
      grad = ctx.createLinearGradient(100, 0, 220, 0);
      grad.addColorStop(0.0, '#000033');
      grad.addColorStop(0.2, '#6699ff');
      grad.addColorStop(0.4, '#003366');
      grad.addColorStop(0.6, '#6699ff');
      grad.addColorStop(1.0, '#000033');
      ctx.fillStyle = grad;
      ctx.fillText('world!', 100, 4);
      grad = ctx.createRadialGradient(227, 15, 5, 235, 25, 24);
      grad.addColorStop(0, 'rgba(198, 198, 255, 1.0)');
      grad.addColorStop(0.9, 'rgba(0, 50, 255, 1.0)');
      grad.addColorStop(1, 'rgba(0, 50, 255, 0)');
      ctx.fillStyle=grad;
      ctx.fillRect(209, 0, 260, 50);"
  />
```

The radial gradient creates both the blue sphere and the glow on it.

More sophisticated shapes, such as the hexagon in the <canvas> demo below, can be created using the methods `beginPath`, `moveTo`, `lineTo` repeated two or more times and `closePath`.

```
<canvas id="ex2canvas" width="200" height="200" style="border: 1px
solid black"></canvas> <iframe src="about:blank" width="0" height="0"
onload="ctx=document.getElementById('ex2canvas').getContext('2d');
ctx.clearRect(0, 0, 200, 200); ctx.beginPath(); ctx.moveTo(148,183.14);
ctx.lineTo(52,183.14); ctx.lineTo(4,100); ctx.lineTo(52,16.86); ctx.lineTo(148,16.86);
ctx.lineTo(196,100); ctx.closePath(); grad = ctx.createLinearGradient(28, 58,
172, 142); grad.addColorStop(0.0, 'rgba(255,0,0,1.0)'); grad.addColorStop(0.25,
'rgba(255,255,0,1.0)'); grad.addColorStop(0.30, 'rgba(0,255,0,1.0)');
grad.addColorStop(0.50, 'rgba(0,255,255,1.0)'); grad.addColorStop(0.66,
'rgba(0,0,255,1.0)'); grad.addColorStop(0.75, 'rgba(255,0,255,1.0)');
grad.addColorStop(1.0, 'rgba(255,0,0,1.0)'); ctx.fillStyle = grad; ctx.fill();"></iframe>
```

Interactive <canvas> tag demo

```
<canvas width="200" height="200" onmouseover="ctx=this.getContext('2d');
   ctx.clearRect(0, 0, 200, 200); ctx.fillStyle='rgba(0,50,10,0.60)';
   ctx.fillRect(0,0,200,200);" onclick="ctx=this.getContext('2d');
   ctx.fillStyle='rgba(0,50,10,0.25)'; ctx.fillRect(50,50,100,100);"
   style="border: 1px solid black; background-color: #ccffcc"
 />
```

Here is an actual working example of the code above.

Mouse over and click the canvas demo below

```
<canvas width="200" height="200" onmouseover="ctx=this.getContext('2d');
ctx.clearRect(0, 0, 200, 200); ctx.fillStyle='rgba(0,50,10,0.60)';
ctx.fillRect(0,0,200,200);" onclick="ctx=this.getContext('2d');
ctx.fillStyle='rgba(0,50,10,0.25)'; ctx.fillRect(50,50,100,100);"
style="border: 1px solid black; background-color: #ccffcc"></canvas>
```

back to top

Changes in HTML 5 - \<canvas\> Tag

What's new in HTML 5

The \<canvas\> tag is one of the new elements in HTML 5.

Differences between HTML 5 and earlier versions of HTML

The \<canvas\> tag did not exist in older versions of HTML.

The 2000-2010 Recommendations from the W3C HTML Working Group defined the HTML namespace for the names of all HTML element types, which now includes the **canvas** element name. In older (pre-2000) versions of HTML, element type names were not associated with a namespace.

back to top

THE END