

Siobhan Lynch

## Dynamic Web1 - Homework 7 part 2

1. Who first introduced canvas and in what year? And what was it used for?

Apple for use in their Mac OS X WebKit component in 2004

it was started being used in Gecko 1.8 (Firefox), and in Opera in 2006, and standardized by WHATWG (Web Hypertext Application Technology Working Group - remember them?) on new proposed specifications for next-generation web technologies.

2. What is canvas 2d and what does 2d stand for?

The canvas tag/element is an HTML5 element that can be used to draw graphics using JavaScript.

The canvas element is only a container for graphics. You must use JavaScript to draw the graphics.

canvas can be used to draw graphs, combine photos, or create animations.

2d stands for two-dimensional graphics

3. How do you define the canvas element?

"<canvas></canvas>", added to the index.html file.

4. The canvas element only supports the use of 2 other attributes aside from the id (or class) attributes. What are they?

Width and height attributes

5. What is the default canvas size (width x height)?

300 x 150

6. If no width or height attribute is added to the canvas element in index.html, and no background, width, or height properties within the canvas element/id in the external stylesheet, or width or height in the JavaScript file, will anything render to the page?

Nothing will render, you will see nothing. You need at least a background to show since the canvas by default has a width and height.

7. If the width and height are set in ctx.fillRect() only, and the width and height were set to greater than 300x x 150h, like 600w x 400h, for example, what width and height of canvas would render to the page?

canvas size of 300w x 150h would be rendered to the page.

8. To get access to the actual drawing interface in canvas, what do we first need to create? And what is that exactly?

we first need to create a context. A context is an object whose methods provide the drawing interface.

9. What are the 2 main drawing styles? What are each for and what are they called?

There are two main drawing styles: "2d" for two-dimensional graphics, and "WebGL" for three-dimensional graphics through the OpenGL interface.

10. How can we access our drawing context once we have retrieved our canvas element by id?

Which built-in HTML5 canvas method does that for us and what is it set on? Give me the variable name and the value it is assigned.

Once we have retrieved the element with the id of "canvas", our canvas element, we can access the drawing context using the HTML5

canvas built-in .getContext() method. So in our index.html, we have <canvas id="canvas"></canvas>.

In addition, the variable name is canvas that given a value of "document.getElementById("canvas").

This is the example that checking for canvas support in the browser:

```
const canvas = document.getElementById("canvas");
let ctx = canvas.getContext("2d");
if (canvas.getContext) {
    ctx = canvas.getContext("2d");
} else {
    const para = document.querySelector(".unsupported");
    para.textContent = `Your browser does not support HTML5 Canvas.`;
}
```

11. What are the two primitive shapes canvas supports?

rectangles and paths.

12. How are all other shapes (other than the primitive rectangle or path) created in canvas?

All other shapes must be created by combining one or more paths.

And what kind of functions help us create complex shapes?

there are a whole bunch of built-in canvas path drawing functions that help us create complex shapes.

13. What are the 3 built-in functions in canvas that draw rectangles in canvas? Tell me the function names and give me the names of the parameters passed into them. Also, describe what each parameter stands for.

The 3 built-in functions in canvas that draw rectangles are:

The first is fillRect().

The parameters passed into this function are "x, y, width, height"

x: the x-axis coordinate of the upper-left corner of the rectangle (the x-axis coordinate starting point).

y: the y-axis coordinate of the upper-left corner of the rectangle (the y-axis coordinate starting point).

width: the rectangle's width. Positive values are to the right. Negative values are to the left.

height: the rectangle's height. Positive values are down. Negative values are up.

The second is strokeRect().

The parameters in this function are "x, y, width, height"

x: the x-axis coordinate of the rectangle's starting point.

y: the y-axis coordinate of the rectangle's starting point.

width: The rectangle's width. Positive values are to the right, and negative values are to the left.

height: the rectangle's height. Positive values are down, and negative values are up.

The third is clearRect().

The parameters in this function are also "x, y,

width, height”

x: the x-axis coordinate of the rectangle's starting point.

y: the y-axis coordinate of the rectangle's starting point.

width: the rectangle's width. Positive values are to the right. Negative values are to the left.

height: the rectangle's height. Positive values are down. Negative values are up.

#### 14. Describe what a path is in canvas.

path: a list of points connected by segments of lines that can be different shapes. They can be curved or not, and of different widths and colors.

A path, or a subpath, can be closed.

#### 15. What are the extra steps to drawing paths in canvas?

Draw a path.

Use drawing commands to draw into the path.

Once a path has been created, `stroke()` or `fill()` the path to render it.