W08 - Canvas, SVG, and Drag and Drop

Canvas - draw through JavaScript. Improve performance by avoiding image transfer off the network. Draw shapes and lines, arcs, text, gradients and patterns.

- Manipulate pixels in images and video.

CANVAS HISTORY -

- Developed by Apple
- Many of the concepts in HTML5 canvas

Creating a Canvas Element

```
<canvas id="myCanvas" class="myCanvas">
  Sorry! Your browser doesn't support Canvas
</canvas>
```

↑ will only display if the browser does not support Canvas

- Canvas elements take height and width attributes
- CSS properties in the stylesheet do not fully implement the size of canvas. The default size of 300 x 150 will display inside the CSS box.

Drawing on the Canvas - happens via the Canvas API

get the Canvas element:

```
const canvas = document.getElementById("myCanvas");
```

INTERFACE: CanvasRenderingContext2D

Call getContext to obtain drawing context.

```
const context = canvas.getContext("2D");
```

↑ instance of CanvasRenderingContext2D

3D is supported using WebGL see:
http://www.khronos.org/webgl/

In08 - Canvas, SVG, and Drag and Drop

Filling Our Brush with Color

strokeStyle or fillStyle properties

- One of three properties: a string representing color, a CanvasGradient object, or a CanvasPattern object.

Style the stroke using a color string:

```
context.strokeStyle = "red";
```

Style the fill using a color string:

```
context.fillStyle = "blue";
```

can use any css color value, as long as it can be specified as a string #00FFFF, red, blue, rgb(0, 0, 255), rgba(0,0,255,0

Now can start drawing:

fillRect and strokeRect methods are used to draw a rectangle using x and y coordinates, and width and height:

```
context.fillRect(10,10,100,100);
context.strokeRect(10,10,100,100);
```

10 px from top and left, draw a rectangle 100px x 100px This will go inside the canvas

Coordinates are 0,0 at the top, left corner

Variations on fillStyle

instead of color, could use CanvasGradient or CanvasPattern

CanvasPattern object will use an image to create a pattern

Drawing other shapes - no built in methods for drawing circles or other shapes.

Paths create a blueprint for lines, arcs, shapes, etc. They are invisible until they are set with strokeStyle, then called fillRect

More Complex Shapes Require 3 Steps

1. layout path
2. stroke path
3. fill path

Canvas, SVG, and Drag and Drop

Draw a circle, using beginPath(). resets the default path to begin a new shape.

Generic function:

```
function drawCircle(canvas) {
    const context = canvas.getContext("2d");
    context.beginPath();
    context.arc(50, 50, 30, 0, Math.PI*2, true);
```

There isn't a circle method, so we use arc to draw a 360° arc.

Signature for arc:

arc(x, y, radius, startAngle, endAngle, anti(clockwise)

X, Y coordinates for beginning of arc. (center of circle)

radius – distance from center to edge

start and end angles along circle's circumference

need to close the path.

```
    context.closePath(); // now stroke and/or fill
    context.strokeStyle = "red";
    context.fillStyle = "blue";
    context.lineWidth = 3;
    context.fill();
    context.stroke();
```

Saving Canvas Drawings - can use toDataUrl method to create a url with an image of the drawing

window.open('url') is deprecated in chrome.

FOUND WORK AROUND USING iframe → see week8.js

Canvas, SVG, and Drag and Drop

Drawing an Image to Canvas

- Redraw an existing image

using the drawImage() method, we can draw an image that exists in a single location, someplace else.

Manipulating Images - More exciting than just making a copy.

getImageData() returns ImageData object.

This contains: width, height, data

- data contains pixel information in the form of an array. Each pixel has 4 values - R, G, B, A
  - A - element's transparency
    - 0 totally transparent
    - 1 opaque

getImageData(0, 0, 1, 1)  one pixel here.

- The array returned will be 4 items long, containing red, green, blue, and alpha for this one pixel.

Security Errors: cannot manipulate images from one domain on another domain. Images to be manipulated must reside on same domain.

Converting an Image from Color to B&W

Once an image is placed onto the canvas, we can use a for loop to iterate through each pixel and change it.

1 → draw image
2 → getImageData
3 → iterate through pixels
4 → putImageData

Canvas, SVG, and Drag and Drop

Manipulating Video with canvas

- Can also use canvas to manipulate video
- Video data
  1. like an image, set up canvas and context.
  2. set up an event listener to react to the play event
  3. draw each frame one at a time
     - load getImageData into variable to avoid reloading for each step.

SVG - scalable vector graphics, allows describing vector graphics using xml

- SVG Images are available via the DOM.
- Technologies such as screen readers can recognise an SVG object through the DOM node.
- can inspect SVG using browser's developer tools
- accessible to search engines

Drawing in SVG

Drawing a circle is pretty easy:

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 400 4
    <circle cx="50" cy="50" r="25" fill="red" />
</svg>
```

DRAG AND DROP - unsupported on Android and iOS

1. Set draggable attribute in HTML on elements
2. Add event listener for dragstart event on any draggable element
3. Add event listener for drag over and drop events on elements to accept dropped items