**JavaScript Drawing Commands (Condensed)**

**HTML Syntax** – Used to link the Javascript to our web page

<!DOCTYPE html>

<html>

 <body>

 <canvas id="a" width="200" height="200">

 This text is displayed if your browser does not support HTML5 Canvas.

 </canvas>

 <script type='text/javascript' src='pie.js'></script>

 </body>

</html>

**Javascript Syntax** – Always start with these first two lines

var a\_canvas = document.getElementById("a");

var draw = a\_canvas.getContext("2d");

**Setting the Color**

draw.fillStyle="#FF0000"; Sets the fill color of the object drawn

draw.strokeStyle="#FF0000"; Sets the color for outlined shapes

**Drawing Lines**

draw.beginPath(); Starts a new series of drawing commands.

draw.moveTo(20, 20); Points to an (X, Y) location within the canvas, not the web page.

draw.lineTo(100, 20); Will draw a line from the current location to the (X, Y) location specified.

draw.stroke(); Actually draws the line(s) that you defined.

draw.closePath(); Ends this group of draw commands. Will cause the last point to reconnect back to the first.

\*If you use several draw.lineTo() commands, you can create a polygon, if you end with a fill() statement.

**Drawing Rectangles**

draw.strokeRect(X, Y, width, height); Creates a rectangle at location (X, Y) with size (width, height) (no fill)

draw.fillRect(20,20,150,100); Same as strokeRect, only the rectangle will be filled in.

draw.clearRect(50, 50, 20, 25); Clears out part of another rectangle

**Drawing Arcs**

draw.beginPath();

draw.arc(X, Y, radius, startAngle, endAngle); Draws an arc centered around point(X, Y) that extends from the starting angle to the ending angle, with a radius of *radius*

draw.stroke(); or draw.fill(); Stroke and fill will have different effects.

\*If you want to draw a pie shape, start with a moveTo() that is the center of the arc.

**Drawing Text**

draw.font = "30px Arial";

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

draw.strokeText(“Hello World”, 10, 50);

**Misc Commands**

Draw.lineWidth=10;