Basic 2D Drawing
JavaScript & HTML 5
Part 2

Thursday September 11, 2014.

Key Concepts

- Draw an arc, form a circle
- More on transformations
  - When is a circle an ellipse
- Object’s own coordinate systems
- If one is good, more is better
  - Replication of objects
- Events, on click and timed events.
The Examples

Circles are arc paths

- Notice the Canvas adjusts to window size.
CSS – Center & % Width

- The canvas is now centered and adjusted

```css
canvas#board {
  background-color: LavenderBlush;
  margin-left: auto;
  margin-right: auto;
  width: 90%;
  display: block;
  padding: 1px;
  border: 1px solid #000000;
}
```

Circles are arc paths

- Much like earlier example, but 3x2 aspect

The width and height are relevant, perhaps best thought of as the size of the canvas bitmap.
Drawing Coordinates?

```javascript
13//
14 * The coordinate system transformations being used deserve some explanation.
15 * The canvas is sized in the HTML to be 300 by 200 pixels.
16 * The set transformation maps this into a new system with
17 * the origin at the lower left and the point 1.5, 1.0 at the upper right.
18 * This is because, by design, height h is less than width w, and so
19 * the vertical axes runs from 0.0 to 1.0 over the extent of the canvas
20 */
21
function init() {
  dboard = document.getElementById("dboard");
  ctxt = dboard.getContext('2d');
  w = dboard.width;
  h = dboard.height;
  ctxt.setTransform(h, 0.0, 0.0, -h, 0.0, h);
  ctxt.lineWidth=0.01;
  drawIt();
}
```

Actually Drawing a Circle

```javascript
1#function drawIt() {
2 drawCircle();
3 }
4
5#function drawCircle() {
6  ctxt.fillStyle = "Yellow";
7  ctxt.strokeStyle = "DarkRed";
8  ctxt.arc(0.75, 0.5, 0.25, 0.0, 2.0*Math.PI);
9  ctxt.fill();
10  ctxt.stroke();
11 }
```

Arguments to arc:
- center horizontal coordinate
- center vertical coordinate
- radius
- start angle for arc in radians
- end angle for arc in radians
There is no draw-ellipse, but

```
5  function drawEllipse()
6      {     
7         ctx.save();
8         ctx.transform(1.5, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 0.0);
9         ctx.fillStyle = "Yellow";
10        ctx.strokeStyle = "DarkRed";
11        ctx.arc(0.75, 0.5, 0.25, 0.0, 2.0*Math.PI);
12        ctx.fill();
13        ctx.stroke();
14     }
```

Scaling moved center of ellipse.

But, this ellipse also appears to have moved to the right.

Operate About the Center

```
5  function drawEllipse()
6      {
7         ctx.save();
8         ctx.transform(1.5, 0.0, 0.0, 1.0, 0.75, 0.5);
9         ctx.fillStyle = "Yellow";
10        ctx.strokeStyle = "DarkRed";
11        ctx.arc(0.0, 0.0, 0.25, 0.0, 2.0*Math.PI);
12        ctx.fill();
13     }
```

Ellipse is now in the center of the canvas.
Replicants

```
function drawIt() {
  drawEllipse("BlueViolet", 0.40, 0.75);
  drawEllipse("Chocolate", 1.10, 0.75);
  drawEllipse("DarkGoldenrod", 0.40, 0.25);
  drawEllipse("Teal", 1.10, 0.25);
}

function drawEllipse(fillColor, cx, cy) {
  ctx.save();
  ctx.transform(1.5, 0.0, 0.0, 1.0, cx, cy);
  ctx.fillStyle = fillColor;
  ctx.strokeStyle = "Black";
  ctx.beginPath();
  ctx.arc(0.0, 0.0, 0.125, 0.0, 2.0*Math.PI);
  ctx.fill();
  ctx.stroke();
  ctx.restore();
}
```

What to Notice

- Matching save and restore calls
  - What changes is later undone
- Words to describe what happens
  - For each of the four calls:
    - Choose specified color
    - Draw circle about origin (0, 0)
    - Scale horizontal dimension by 1.5
    - Translate to final desired position
An ‘object’

```
function drawPacman(cx, cy) {
    ctx.fillStyle = "Yellow";
    ctx.beginPath();
    ctx.arc(0, 0, 0.25 * Math.PI, 1.25 * Math.PI, false);
    ctx.fill();
    ctx.beginPath();
    ctx.arc(0, 0, 0.75 * Math.PI, 1.75 * Math.PI, false);
    ctx.fill();
    ctx.beginPath();
    ctx.fillStyle = "Black";
    ctx.arc(0.125, 0.125, 0.000 * Math.PI, 2.000 * Math.PI, false);
    ctx.fill();
    ctx.restore();
}
```
What to notice

- Rotation is a common local operation
  - Think about term 'local'!

- Events and events handling are key
  - Some events come from a mouse
  - Some from a clock

- On the web, take care with animation
  - It is a sharp knife – and not always appreciated

The End