

Computer Programming

Canvas Final Pt I – Bubbles (Stationary)

Objective: Use the HTML5 canvas object and Javascript to create a bubbles type screen-saver. You will need to understand how to create a random number within a range.

Suppose you'd like a random integer between *min* and *max*. The formula is this:

$$\text{Math.floor}(\text{Math.random}() * (\text{max}-\text{min}+1)) + \text{min}$$

As an example, if you want a random integer between 50 and 80 you would use this calculation:

$$r = \text{Math.floor}(\text{Math.random}() * 31) + 50$$

1. In your *Computer Programming* folder, create a folder titled *first_last_canvas_final*. Copy the *template.html* file and paste it in your *first_last_canvas_final* folder. Rename the *template.html* file to *bubbles.html*.
2. Edit the *bubbles.html* file,
 - a) Change *CHANGE ME* to *First Last – Canvas Final*
 - b) Change the width of `<canvas>` to 500 and the height to 500.
 - c) In the `draw()` method, draw 30 circles whose location, radius, and color are generated at random. To do this, you will create *Circle* function similar to the one we did in *mouse_events*. You do not need to accept any parameters as you will generate random values when the *Circle* is created. You need to keep track of the following variables and functions:

this.x

this.y

this.radius

this.color

this.draw()

- d) First, the radius should be a random integer between 25 and 55.
- e) Next, create random integers for x and y so that a circle drawn with (x,y) as the center will be fully visible on the canvas. Ask yourself, what are the minimum values for x and y? What are the maximum values for x and y? You must use *can.width* and *can.height* in your calculations. If I change the size of your canvas, your circles must still draw completely on the canvas.
- f) You are now to create a random color. To do this, you will use 4 variables: one for *red*, one for *green*, one for *blue*, and one for *alpha*. Each of the *red*, *green* and *blue* variables will contain a random integer between 40 and 230. The variable for alpha is simply the value *Math.random()*. (This will give a value between 0 and 1). You will set *this.color* to a string that uses the *rgba* function and the concatenation of your variables as follows:

```
this.color = "rgba(" + red + "," + green + "," + blue + "," + alpha + ")";
```

- g) The `draw()` function should draw the circle and fill the color. (Look at the *Circle* function in *mouse_events*).
- h) You need a global variable named *bubbles*. In the `init()` function, you will *initialize* bubbles to be a new array. (Look at how the variable *shapes* was initialized in *mouse_events*.)

- i) After you have initialized the *bubbles* variable, you are to add 30 circles to the array. This is done by creating a new *Circle* and then pushing it onto *bubbles*. (Again, look in *mouse_events* for samples.)
- j) The *draw()* function should simply loop through the *bubbles* array and draw each circle. *mouse_events* has an example of this as well.



3. Zip your *first_last_canvas_final* folder. Turn in your *first_last_canvas_final.zip* file.