Ziggy Sheynin

Mr. Ettlin

APCSP, Period 1

20 January 2020

Lab 130: ArtTwo Code

Index

```
<!DOCTYPE html>

<html>

  <head>

    <meta charset="UTF-8">

    <title>Art Two</title>

    <script src="libraries/p5.js"

type="text/javascript"></script>

    <script src="libraries/p5.dom.js"

type="text/javascript"></script>

    <script src="libraries/p5.sound.js"

type="text/javascript"></script>

    <script src="sketch.js" type="text/javascript"></script>

    <script src="boids.js" type="text/javascript"></script>

    <style> body {padding: 0; margin: 0;} canvas

{vertical-align: top;} </style>

  </head>

  <body>
```

```
  </body>
</html>
```

Sketch

```
//  Ziggy Sheynin
//   Lab 130- Art Two
//  This is a comment
//  The setup function function is called once when your program
begins


var circle = 100;
var rot;
var color;
var freq = 0.000001;
var cont = 0;
var r;
var boids = [];
function setup() {
  var cnv = createCanvas(800, 800);
  cnv.position((windowWidth-width)/2, 30);
  background(5, 5, 5);
  loadBoids(30); //calls function loadBoids
}
//  The draw function is called @ 30 fps
```

```
function draw() {
  background(5);
  translate(width/2, height/2);
  rotate(radians(rot));
  ellipseMode(CENTER);
    for (var i=0; i<500; i++){
      circle = 50 + 85*sin(millis()*freq*i);
      color = map(circle,150,250,255,60);
      r = map(circle,150,250,5,2);
      fill(color,0,74);
      //noStroke();
      ellipse(circle*cos(i), circle*sin(i),r,r);
      rot=rot+0.00005;
    } //end for loop
      runBoids();
  }//end draw
function loadBoids(n){
  for (var i =0; i< n; i++){
   boids[i] = new Boid(5,5,5,5); //initializes and declares boid
objects
    }
  }
function runBoids(){
```

```
  for (var i =0; i<boids.length; i++){
    boids[i].run();
    }
}//++++++++++++++++++++++++++end sketch
```

Boids

```
//Ziggy Sheynin
//Lab 130 Art Two
//This is a comment
//
//
class Boid{
  constructor (x, y, dx, dy, clr){
    this.loc = createVector(x, y);
    this.vel = createVector(dx, dy);
    this.acc = createVector(0, 1);
    this.clr = color(random(255), random(255), random(255));
  } //end Boids constructor
  run(){
    this.render();
    this.checkEdges();
    this.update();
  } //end run function
```

```
  update(){
      this.vel.limit(3);
      this.loc.add(this.vel);
      this.vel.add(this.acc);
  }//end update
  checkEdges(){ //keeps boids on screen
    if(this.loc.x<0){
      this.loc.x=width;
    }
    if(this.loc.x>width){
      this.loc.x=0;
    }
    if(this.loc.y<0){
       this.loc.y=height;
    }
    if(this.loc.y>height){
       this.loc.y=0;
    }
  }//end checkEdges
//  Either warp or bounce
  render(){
    for (var i=boids.length-1; i >0; i--){ //for loop to
traverse array
```

```
      if(this.loc.dist(boids[i].loc)<200){ //checks the distance
between two boids
        stroke(255);
        fill(30, 40, 150, 40); //gives the squares a color
        ellipse(random(0,800), random(0,800), 10, 10); //draws
square
      }
    }
  }//end render
//This method will draw a line between its location and the
location of any other boid object within 200px.
}//++++++++++++++++++++++++++++++++++End Boid Class
```