Object Oriented Program

lbg@dongseo.ac.kr

initial code

```
float x=100, y=100, hue=60, radius=100;
void setup() {
  size(500, 500);
  rectMode(CENTER);
  noStroke();
  smooth();
  colorMode(HSB, 360, 100, 100);
}

void draw() {
  background(255);
  translate(x, y);

  noStroke();
  fill(hue, 100, 100);
  ellipse(0, 0, radius, radius);
}
```

```
float x=100, y=100, hue=60, radius=100;
void setup() {
  size(500, 500);
  rectMode(CENTER);
  noStroke();
  smooth();
  colorMode(HSB, 360, 100, 100);
}

void draw() {
  background(255);
  translate(x, y);

  noStroke();
  fill(hue, 100, 100);
  ellipse(0, 0, radius, radius);
}
```

class code

```
class MovingMark {
  float x=100, y=100, hue=60, radius=100;
  MovingMark() {
    x = random(width);
    y = _________________(1)
    hue = _________________(2)
    radius = random(50, 100);
  }
  MovingMark(float x, float y, float hue, float radius) {
    ____________(3)
    this.y = y;
    this.hue = hue;
    this.radius = radius;
  }
  void draw() {
    translate(x, y);
    noStroke();
    fill(hue, 100, 100);
    ellipse(0, 0, radius, radius);
  }
}
```

change main code - variable style

```
MovingMark mk1, mk2, mk3;

void setup() {
  size(500, 500);
  noStroke();
  smooth();
  colorMode(HSB, 360, 100, 100);
  mk1 = new MovingMark();
  mk2 = new MovingMark(100, 100, 60, 80);
  mk3 = new MovingMark(200, 300, 120, 120);
}

void draw() {
  background(255);
  mk1.draw();
  mk2.draw();
  mk3.draw();
}
```

```
class MovingMark {
  float x=100, y=100, hue=60, radius=100;
  MovingMark() {
    x = random(width);
    y = random(height);
    hue = 60;
    radius = 100;
  }
  MovingMark(float x, float y, float hue, float radius) {
    this.x = x;
    this.y = y;
    this.hue = hue;
    this.radius = radius;
  }
  void draw() {
    translate(x, y);
    noStroke();
    fill(hue, 100, 100);
    ellipse(0, 0, radius, radius);
  }
}
```

change main code - array style

```
MovingMark[] mks = new MovingMark[20];
void setup() {
  size(500, 500);
  noStroke();
  smooth();
  colorMode(HSB, 360, 100, 100);
  for(int i=0; i<mks.length; i++)
    mks[i] = new MovingMark();
}
void draw() {
  background(255);
  drawAxis();
  for(int i=0; i<mks.length; i++) {
    mks[i].draw();
    mks[i].update();
  }
}
void drawAxis() {
  for(int x=0; x<width; x+=100)
    line(x, 0, x, height);
  for(int y=0; y<height; y+=100)
    line(0, y, width, y);
}
```

```
class MovingMark {
  float x=100, y=100, hue=60, radius=100;
  MovingMark() {
    x = random(width);
    y = random(height);
    hue = random(360);
    radius =  random(50, 100);
  }
  MovingMark(float x, float y, float hue, float radius) {
    this.x = x;
    this.y = y;
    this.hue = hue;
    this.radius = radius;
  }
  void draw() {
    pushMatrix();
    translate(x, y);
    fill(hue, 100, 100);
    ellipse(0, 0, radius, radius);
    popMatrix();
  }
  void update() {
    hue++;
    if(hue>360) hue=0;
    x+=random(-5,5);
    y+=random(-5,5);
  }
```

change main code - arraylist style

```
ArrayList<MovingMark> mks;
void setup() {
  size(500, 500);
  colorMode(HSB, 360, 100, 100);
  mks= new ArrayList<MovingMark>();
  mks.add(new MovingMark());
}
void draw() {
  background(255);
  drawAxis();
  for(int i=0; i<mks.size(); i++) {
    MovingMark mk = mks.get(i);
    mk.draw();
    mk.update();
  }
}
void mousePressed() {
  mks.add(new MovingMark(mouseX, mouseY, random(360),
random(50,100)));
}
void drawAxis() {
  for(int x=0; x<width; x+=100)
    line(x, 0, x, height);
  for(int y=0; y<height; y+=100)
```

```
class MovingMark {
  float x=100, y=100, hue=60, radius=100;
  MovingMark() {
    x = random(width);
    y = random(height);
    hue = 60;
    radius = 100;
  }
  MovingMark(float x, float y, float hue, float radius) {
    this.x = x;
    this.y = y;
    this.hue = hue;
    this.radius = radius;
  }
  void draw() {
    pushMatrix();
    translate(x, y);
    noStroke();
    fill(hue, 100, 100);
    ellipse(0, 0, radius, radius);
    popMatrix();
  }
}
```