Chapter 2 Recap

• basic drawing commands
  – rect(), line(), size(), background(), stroke(),...

• function call statements
  – rect(10,10,40,60);

• working with the Processing environment

• Comments
Chapter 3

• draw()
• setup()
• mouseX, mouseY
• frameRate()
// A circle following the mouse leaving a trail behind. Take
// note of draw and setup – these are new.

void setup() {
    size(400, 400);
}

draw() {
    ellipse(mouseX, mouseY, 10, 10);
}
// A circle following the mouse, not leaving a trail behind

void setup() {
    size(400, 400);
}

void draw() {
    background(150);
    ellipse(mouseX, mouseY, 10, 10);
}
// Will this leave a trail of circles (A) or show just one (B)?
void setup() {
    size(400, 400);
    background(255, 0, 0);
}
void draw() {
    ellipse(mouseX, mouseY, 10, 10);
}
Any Visible difference in output? (A) for Yes, (B) for No

/**
* Draw a rudimentary car.
* @author Dustin Adams
*/

void setup() {
    size(400,400);
}

void draw() {
    // draw the body
    rect(100,100,200,50);
    // draw the wheels
    ellipse(150,150,50,50);
    ellipse(250,150,50,50);
    // draw the windshield
    line(150,100,200, 50);
}
// moving the car around
void setup() {
  size(400,400);
}

void draw() {
  background(100);
  // draw the body
  rect(mouseX, mouseY, 200, 50);
  // draw the wheels
  ellipse(mouseX+50, mouseY+50, 50, 50);
  ellipse(mouseX+150, mouseY+50, 50, 50);
  // draw the windshield
  line(mouseX+50, mouseY, mouseX+100, mouseY-50);
}
// stroke thickness based on mousex. Taken from exercise 3-7 in // Learning Processing

void setup() {
    size(400, 400);
}

void draw() {
    strokeWeight(abs(mouseX-pmouseX));
    line(pmouseX, pmouseY, mouseX, mouseY);
}
// similar to example from a few slides ago, but this changes // background based on user input.

void setup() {
    size(400, 400);
    background(175);
}

void draw() {
    strokeWeight(abs(mouseX-pmouseX));
    line(pmouseX, pmouseY, mouseX, mouseY);
}

void mousePressed() {
    background(255,0,0);
}

void keyPressed() {
    background(175);
}

void keyReleased() {
}
// frameCount keeps a running total of how many frames have
// been shown during the life of the sketch.
// This sketch shows an ellipse moving across the screen
// following the frameCount.

void setup() {
    size(400,300);
}
void draw() {
    background(255);
    ellipse(frameCount, 150, 40, 40);
    println(frameCount);
}

// This sketch shows a circle growing in size as the frameCount increases

void setup() {
  size(400, 300);
}

void draw() {
  background(255);
  ellipse(200, 150, frameCount, frameCount);
  println(frameCount);
}
// The beginnings of a sunset
void setup() {
    size(400, 300)
    fill(255, 255, 0); // yellow for the sun
    background(175, 212, 255); // light blue sky
    noStroke();
}
draw() {
    Ellipse(200, frameCount, 40);
}
How many mistakes can you find in this program?
A: 1–2
B: 3
C: 4
D: 5
E: more than 5
// Default frameRate is 30 frames per second. This example slows it down to 1 frame per second.

void setup() {
  frameRate(1);
}
void draw() {
  background(100);
  ellipse(mouseX, mouseY, 30, 30);
}
// Example 3-5 from Learning Processing

global void setup() {
    size(200, 200);
    background(255);
}

global void draw() {
}

global void mousePressed() {
    stroke(0);
    fill(175);
    rectMode(CENTER);
    rect(mouseX, mouseY, 16, 16);
}

global void keyPressed() {
    background(255);
}
// Example 3-6 from Learning Processing

void setup() {
    // Set the size of the window
    size(200, 200);
    frameRate(30);
}

void draw() {
    // Draw a white background
    background(255);
    // Set ellipses and rects to CENTER mode
    ellipseMode(CENTER);
    rectMode(CENTER);
    // Draw Zoog's body
    stroke(0);
    fill(175);
    rect(mouseX, mouseY, 20, 100);
    // Draw Zoog's head
    stroke(0);
    fill(255);
    ellipse(mouseX, mouseY-30, 60, 60);
    // Draw Zoog's eyes
    fill(mouseX, 0, mouseY);
    ellipse(mouseX-19, mouseY-30, 16, 32);
    ellipse(mouseX+19, mouseY-30, 16, 32);
    // Draw Zoog's legs
    stroke(0);
    line(mouseX-10, mouseY+50, pmouseX-10, pmouseY + 60);
    line(mouseX+10, mouseY+50, pmouseX+10, pmouseY + 60);
}
void mousePressed() {
    println("Take me to your leader!");
}

Summary

- Summary/Review
- coordinate system
- basic drawing commands and their parameters (rect, line, ellipse, background, stroke, fill)
- color model - RGB + alpha
- Processing IDE - entering/saving/running
- top to bottom statement execution - order matters
- function call syntax (e.g. rect(10,20,30,40); )
- println() for debugging
- comments
- setup & draw
- mouseX, pmouseY
- Did I leave off something important?