PUCD 2035
CORE 1 INTERACTION

SECTION B
T 9–11h40

SESSION TWELVE
NOVEMBER 14, 2023

P5: 3D + 4D

DAVIS SCHERER
SCHERERD@NEWSCHOOL.EDU

AGENDA

1. COMING UP
2. MORE P5
3. HOMEWORK

COMING UP

11/14	11/21	11/28	12/5
P5	Metadata + Access	Review + Wrap Up	FINAL Critiques

FINALS

Finals are in three weeks!

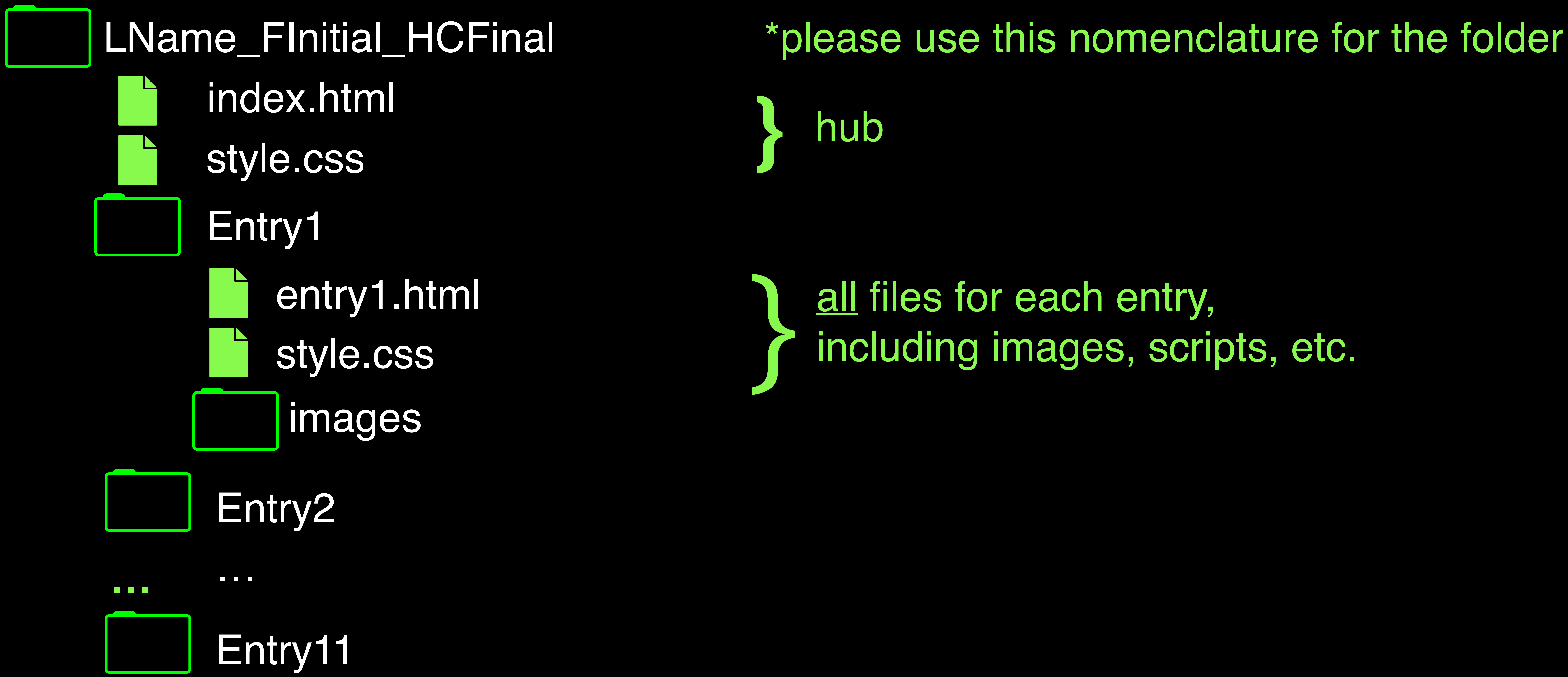
1. FILES

Please submit **FILES** (not links) of your **ENTIRE, FULLY LINKED, FULLY FUNCTIONAL** collection.

1. FILES

You are very welcome to host your files on GitHub, but I **ALSO** need a full package organized as follows submitted as **actual files** to Canvas.

1. FILES



- LName_FInitial_HCFinal
  - index.html
  - style.css
  - Entry1
    - entry1.html
    - style.css
    - images
  - Entry2
  - ...
  - ...
  - Entry11

*please use this nomenclature for the folder

} hub

} all files for each entry,
including images, scripts, etc.

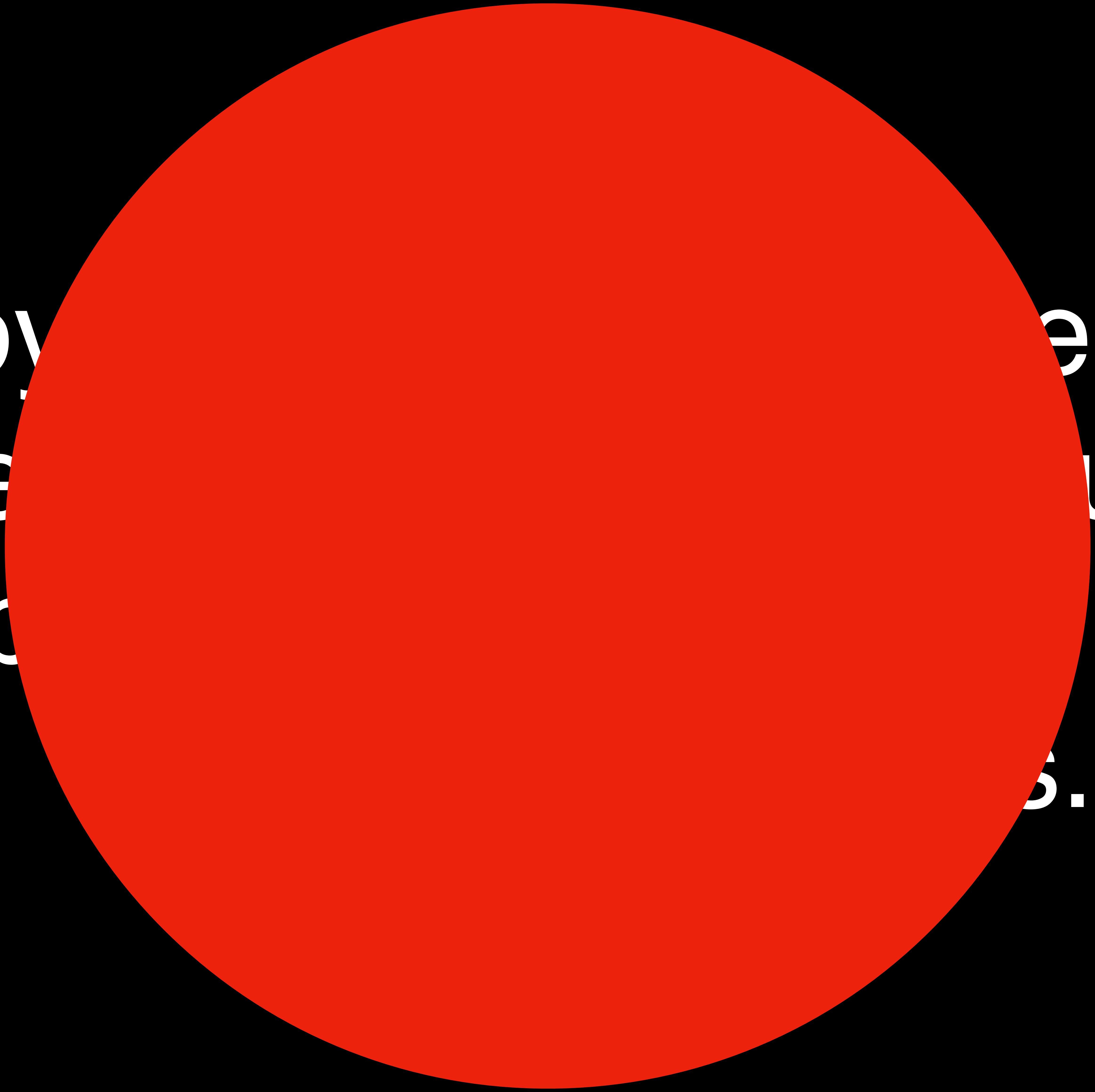
2. CRITIQUE

We will run the critique just like the midterm— be prepared to present off of your computer, and to respond to your classmates' work.

FINALS

QUESTIONS?

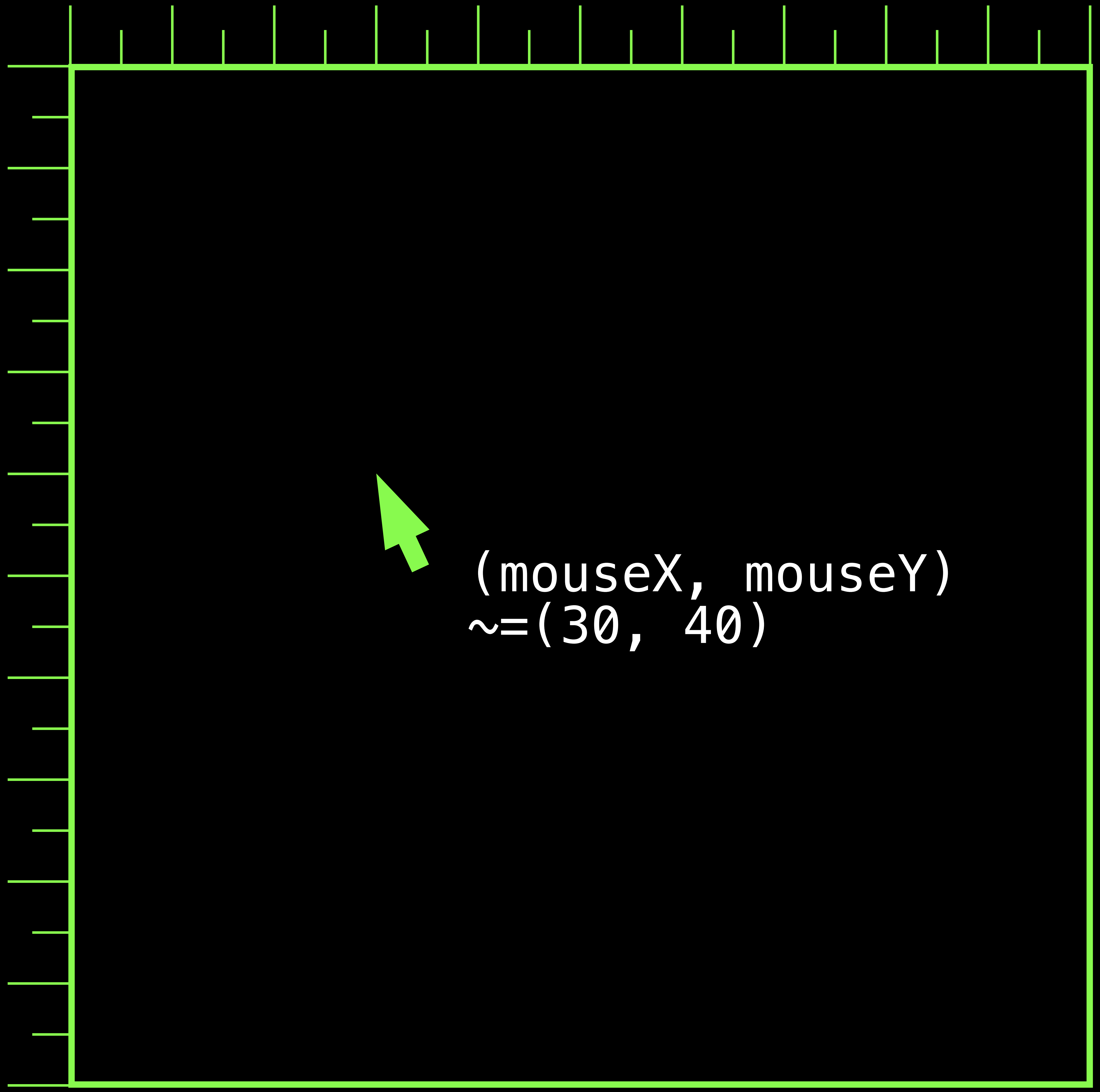
INTERACTIVITY IN P5



You toy eractivity
last we unctions
and inp may not
s.

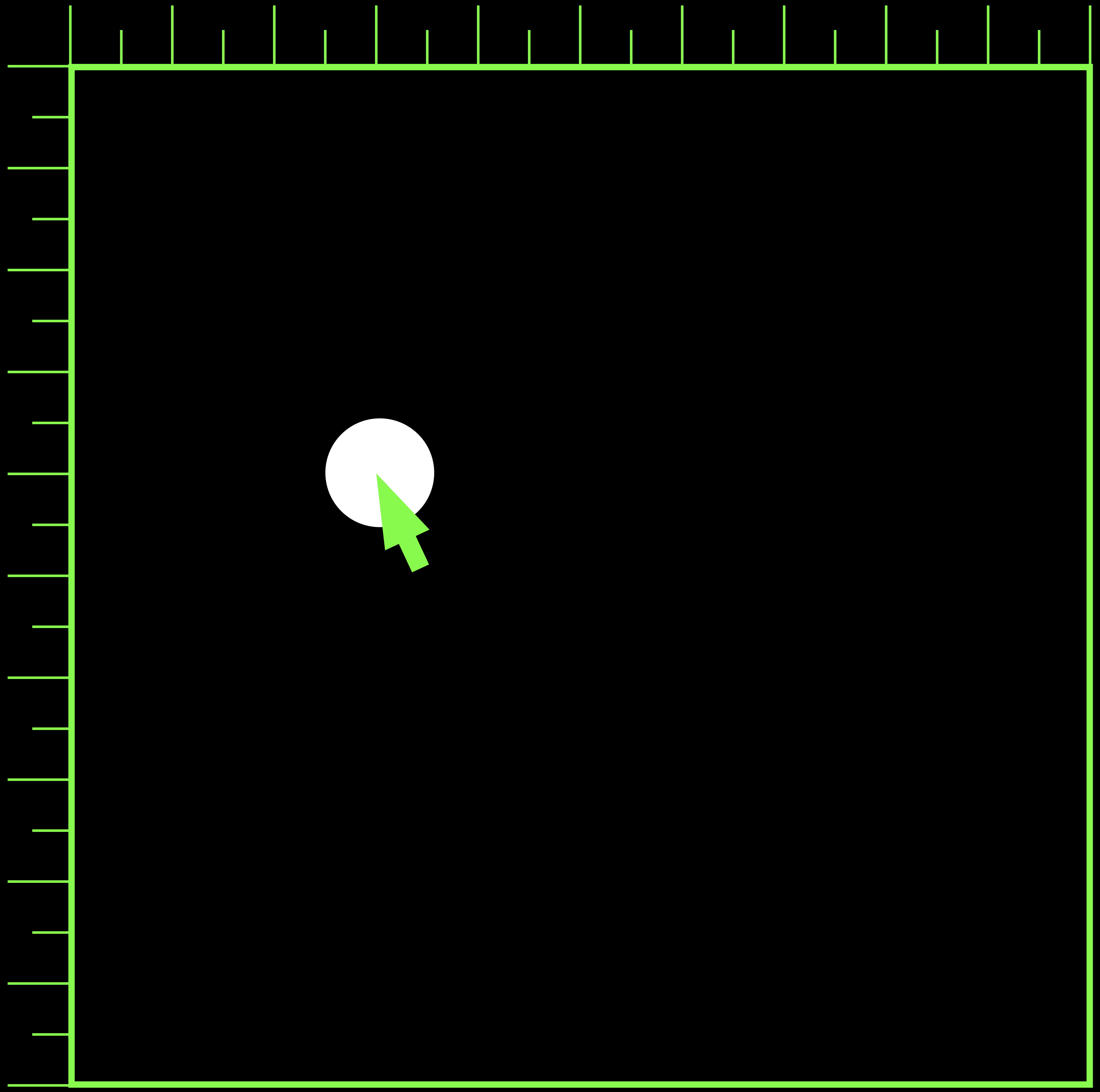
MOUSE DATA

```
function setup() {
  createCanvas(100, 100);
}
```



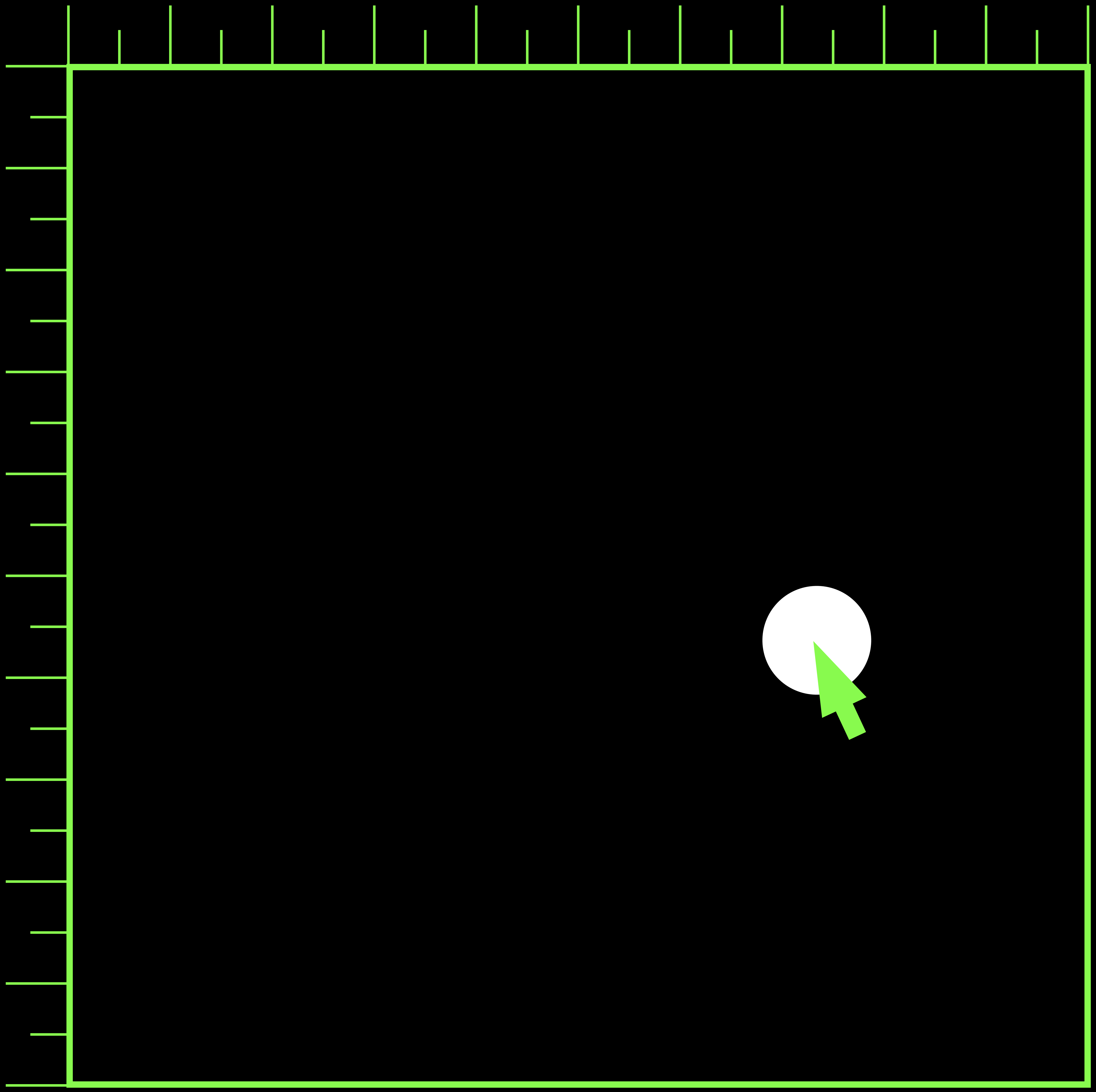
MOUSE DATA

```
function setup() {
  createCanvas(100, 100);
}

function draw() {
  fill(255);
  ellipse(mouseX, mouseY, 10, 10);
}
```



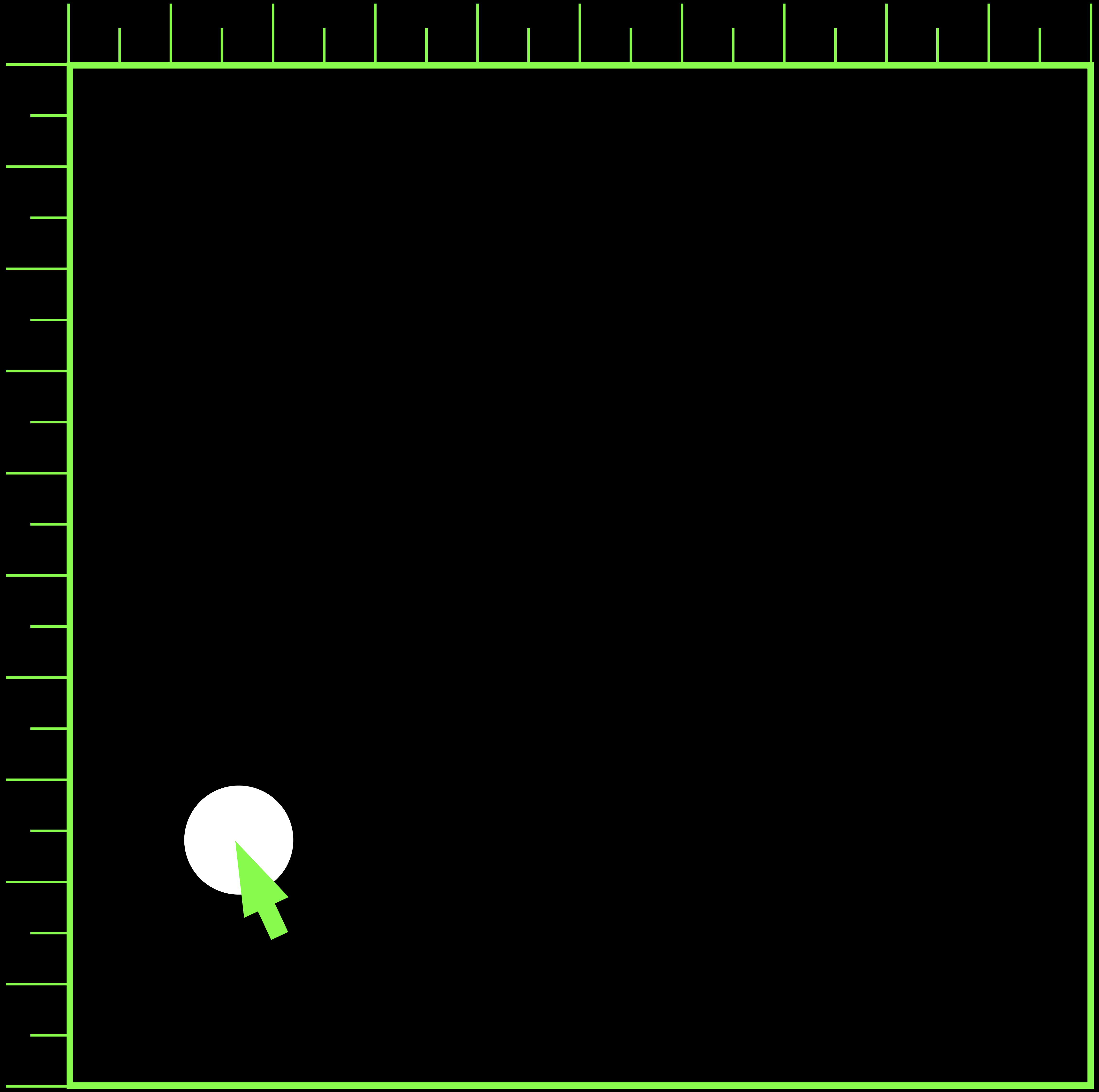
MOUSE DATA

```
function setup() {
  createCanvas(100, 100);
}

function draw() {
  fill(255);
  ellipse(mouseX, mouseY, 10, 10);
}
```



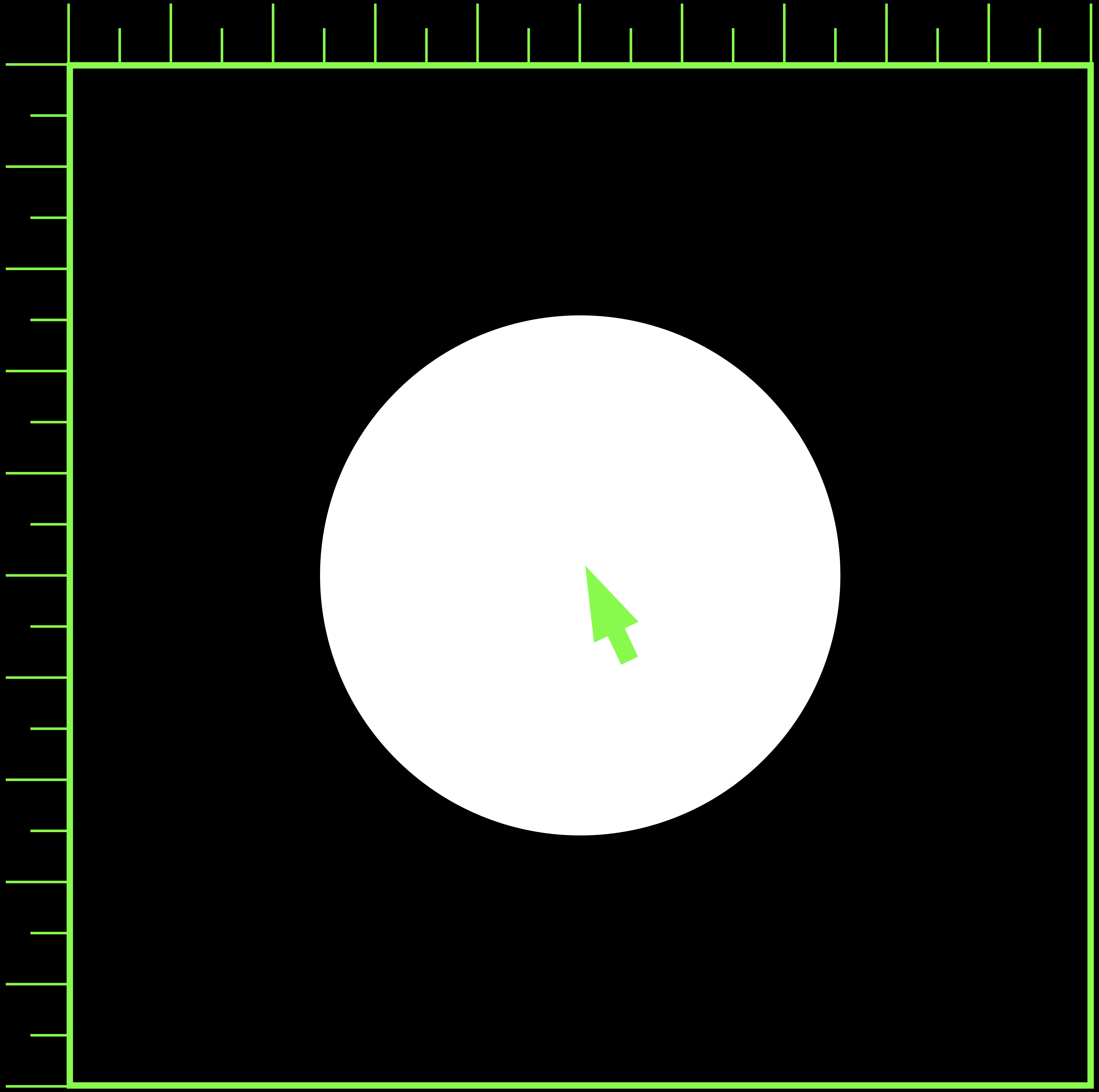
MOUSE DATA

```
function setup() {
  createCanvas(100, 100);
}

function draw() {
  fill(255);
  ellipse(mouseX, mouseY, 10, 10);
}
```



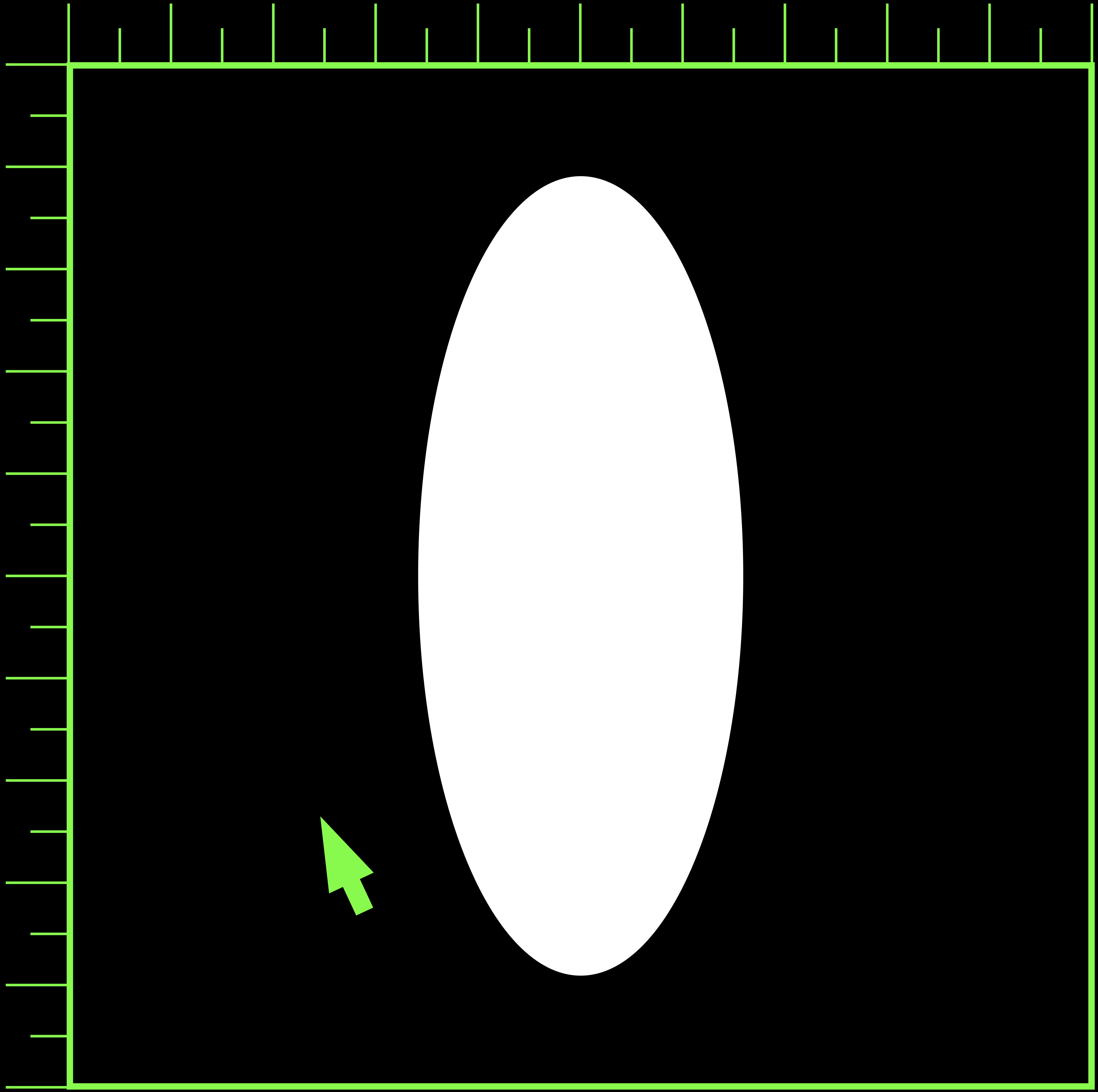
MOUSE DATA

```
function setup() {
  createCanvas(100, 100);
}

function draw() {
  fill(255);
  ellipse(50, 50, mouseX, mouseY);
}
```



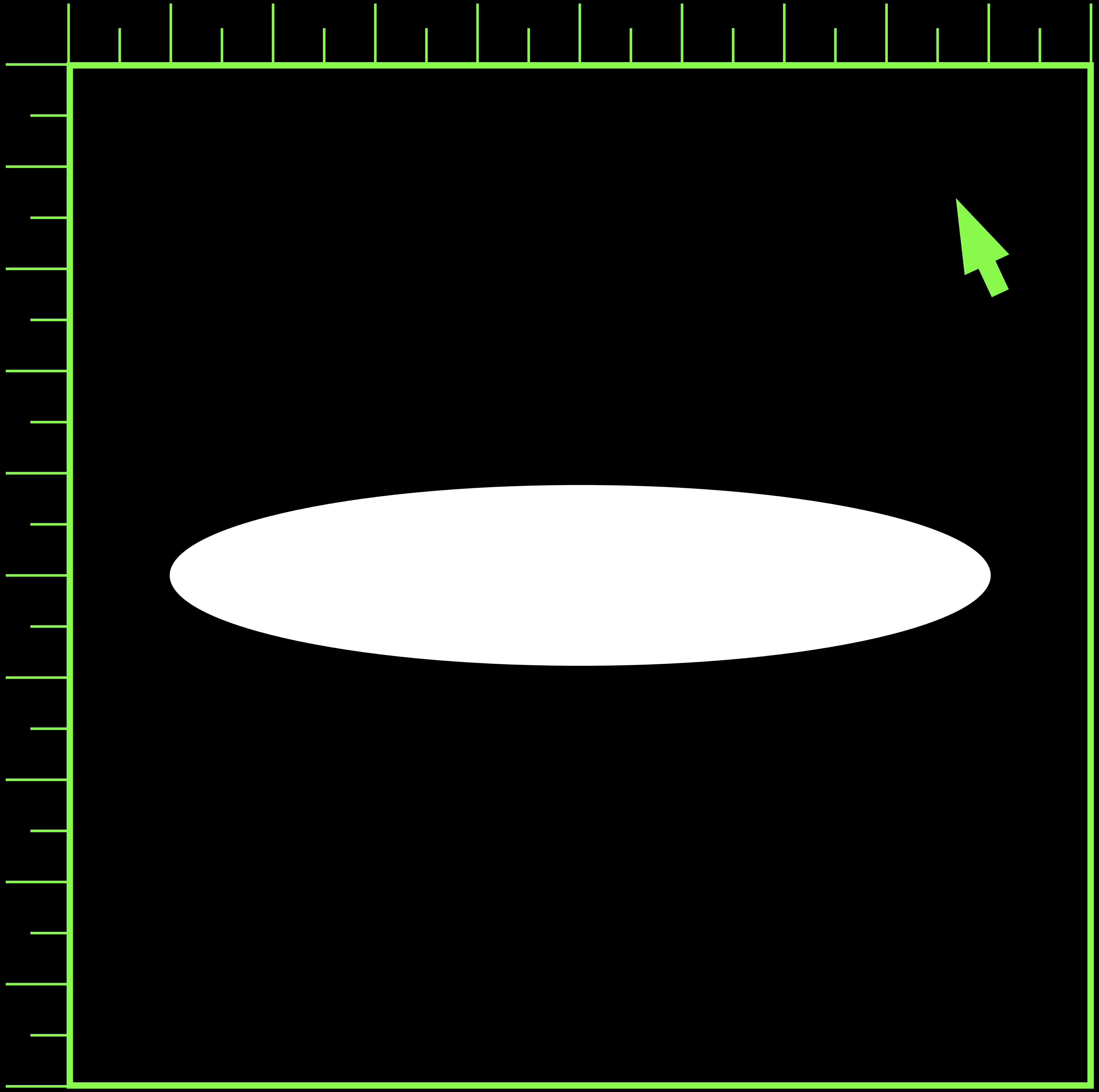
MOUSE DATA

```
function setup() {
  createCanvas(100, 100);
}

function draw() {
  fill(255);
  ellipse(50, 50, mouseX, mouseY);
}
```



MOUSE DATA

```
function setup() {
  createCanvas(100, 100);
}

function draw() {
  fill(255);
  ellipse(50, 50, mouseX, mouseY);
}
```



INVERTING MOUSE DATA

```
function setup() {
  createCanvas(100, 100);
}

function draw() {
  let inverseX = (width-mouseX);
  let inverseY = (height-mouseY);
  fill(255);
  ellipse(mouseX, mouseY, inverseX,
  inverseY);
```

INVERTING MOUSE DATA

```
function setup() {
  createCanvas(400, 400);
}

function draw() {
  let inverseX = (width-mouseX);
  let inverseY = (height-mouseY);
    background(255,0,0);
  fill(255,255,0);
    noStroke();
  ellipse(mouseX, mouseY, inverseX, inverseY);
}
```

MOUSE DATA AND CONDITIONALS

```
function setup() {
  createCanvas(100, 100);
  noStroke();
  fill(0);
}
function draw() {
  background(255);
  if (mouseX < 33) {
    fill(255,0,0);
    rect(0, 0, 33, 100);  // Left
  }
  else if (mouseX < 66) {
    fill(0,255,0);
    rect(33, 0, 33, 100); // Middle
  }
  else {
    fill(0,0,255);
    rect(66, 0, 33, 100); // Right
  }

  if(mouseY < 33){
    fill(255,255,0,127);
    fill
    rect(0,0,100,33);
  }
  else if(mouseY < 66){
      fill(255,255,0,127);
  rect(0,33,100,33);
  }
  else{
     fill(255,255,0,127);
  rect(0,66,100,33);
  }

}
```

MOUSE PRESSES

```
function setup() {
  createCanvas(400, 400);
  noStroke();
}
function draw() {
  background(0,0,255);
  if (mouseIsPressed == true) {
    fill(255,255,0);
  }
  else {
    fill(255,0,0);
 }
    rect(25, 25, 60, 350);
    rect(25, 175, 350, 50);
    rect(315, 25, 60, 350);
}
```

KEY PRESSES

```
let x = 20;
function setup() {
  createCanvas(100, 100);
  strokeWeight(4);
  stroke(255)
}
function draw() {
  background(255,0,0);
  if (keyIsPressed == true) {
    x++;
  }

  if (x >=width){
    x = 0;
  }
  ellipse(x, 20, 20, 20);
  ellipse(x, 40, 20, 20);
  ellipse(x+20, 20, 20, 20);
  ellipse(x+20, 40, 20, 20);
}
```

KEY PRESSES

```
let x = 20;
function setup() {
  createCanvas(100, 100);
  strokeWeight(4);
  stroke(255)
}
function draw() {
  background(255,0,0);
  if ((keyIsPressed == true) && (key == 't')){
    x++;
  }

  if (x >=width){
    x = 0;
  }
  ellipse(x, 20, 20, 20);
  ellipse(x, 40, 20, 20);
  ellipse(x+20, 20, 20, 20);
  ellipse(x+20, 40, 20, 20);
}
```

EVENTS

You can also call actions to happen on a specific event using the format *`function eventName()`*

EVENTS

mousePressed()

mouseReleased()

doubleClicked()

mouseMoved()

mouseDragged()

mouseOver()

mouseOut()

EVENTS

```
let x = 100;
let y = 100;

function setup() {
  createCanvas(300, 300);
}
function draw() {
  background(255,0,0);
  fill(255,255,0);
  noStroke();
  rect(x,y,100,100);

}

function mouseClicked(){
    x = mouseX;
    y = mouseY;
  }
```

1:

Make a sketch where something changes based on user input—mouse click, keyboard stroke, etc.

1:

(SHARE)

TIME IN P5

P5 can access the system clock to use live time as a variable for display or computation.

TIME IN P5

```
let y = year();
let d = day();
let h = hour();
let m = minute();
let s = second();
let ms= millis();
//you need to store the time in a variable
so you can use it in your sketches.
```

TIME IN P5

```
function setup(){
  createCanvas(200,200);
}

function draw(){
  let h = hour();
  let m = minute();
  let s = second();

  let hourR = (h/24)*255;
  let minuteG = (m/60)*255;
  let secondB = (s/60)*255;

  background(hourR, minuteG, secondB);
  text('The time is' + h + ':' + m + ':' + s, 50, 50);
  }
```

TIME IN P5

```
function setup(){
  createCanvas(200,200);
}

function draw(){
  let h = hour();
  let m = minute();
  let s = second();

  let locX = (s/24)*200;
  let locY = (m/60)*100;
  let sizeS = (s/60)*200;

  background(0,0,255);
  fill(255,255,0);
  noStroke();
  ellipse(locX, locY, sizeS, sizeS);
  }
```

RANDOM

P5 can generate random variables for you to use, if you want or need.

RANDOM

```
function setup(){
  createCanvas(200,200);
}

function draw(){
  let x = random(width);
  let y = random(height);

  frameRate(5);

  fill(255,0,0);
  noStroke();
  background(255,255,0);
  ellipse(x, y, 10, 10);
}
```

RANDOM

```
function setup(){
  createCanvas(200,200);
}

function draw(){
  let flags = ['🏴', '🏁', '⛳'];
  let flag = random(flags);

  frameRate(1);

  textSize(100);
  text(flag, 100, 100);
}
```

2:

Make a sketch where something
is dictated based on either a random input
or the use of time loaded in as a variable.

2:

(SHARE)

3D IN P5 (WEBGL)

P5 can also create a 3D space to sketch within.

3D IN P5 (WEBGL)

To use webGL, you have to declare it in the setup as part of the canvas.

```
createCanvas(width, height, WEBGL)
```

There are several 3D primitives.
plane, box, sphere, cylinder,
cone, ellipsoid, torus

3D: MAKING A SHAPE

```
function setup() {
  createCanvas(100, 100, WEBGL);
}

function draw() {
  background(200);
  rotateX(frameCount * 0.01);
  rotateY(frameCount * 0.01);
  box(50);
}
```

3D: MATERIAL

```
function setup() {
  createCanvas(100, 100, WEBGL);
 normalMaterial();
}

function draw() {
  background(200);
  rotateX(frameCount * 0.01);
  rotateY(frameCount * 0.01);
  box(50);
}
```

3D: LIGHTING

```
function setup() {
  createCanvas(100, 100, WEBGL);
  noStroke();
}

function draw() {
  background(0);

  ambientLight(60);

  let locX = mouseX - width / 2;
  let locY = mouseY - height / 2;
  pointLight(255, 255, 255, locX, locY, 50);

  specularMaterial(250);
  shininess(50);
  torus(30, 10, 64, 64);
}
```

3D: INTERACTIVITY

```
function setup() {
  createCanvas(100, 100, WEBGL);
  noStroke();
}

function draw() {
  background(0);

  ambientLight(60);

  let locX = mouseX - width / 2;
  let locY = mouseY - height / 2;
  pointLight(255, 255, 255, locX, locY, 50);

  specularMaterial(250);
  shininess(50);

 orbitControl();

  torus(30, 10, 64, 64);
}
```

3D: INTERACTIVITY

```
function setup() {
  createCanvas(300, 300, WEBGL);
  noStroke();
    detailY = createSlider(3, 16, 3);
  detailY.position(10, height + 5);
  detailY.style('width', '80px');

}

function draw() {
  background(0);

  ambientLight(60);

let locX = mouseX - width / 2;
  let locY = mouseY - height / 2;
  pointLight(255, 255, 255, locX, locY, 50);

  specularMaterial(250);
  shininess(50);

 orbitControl();

  sphere(40, 16, detailY.value());
}
```

3:

Make a sketch in WEBGL. It can be static, it can move… Whatever you want.

BREAK (10 M)

FOR THE REST OF CLASS:

Sketch around in P5. It can be related to your HC, it can be related to something in another class, it can just be for fun.

Email me your sketch before you go.

HOMEWORK

1. HC 10
2. Start thinking about your final collection.

SESSION TWELVE
NOVEMBER 14, 2023

THANK YOU