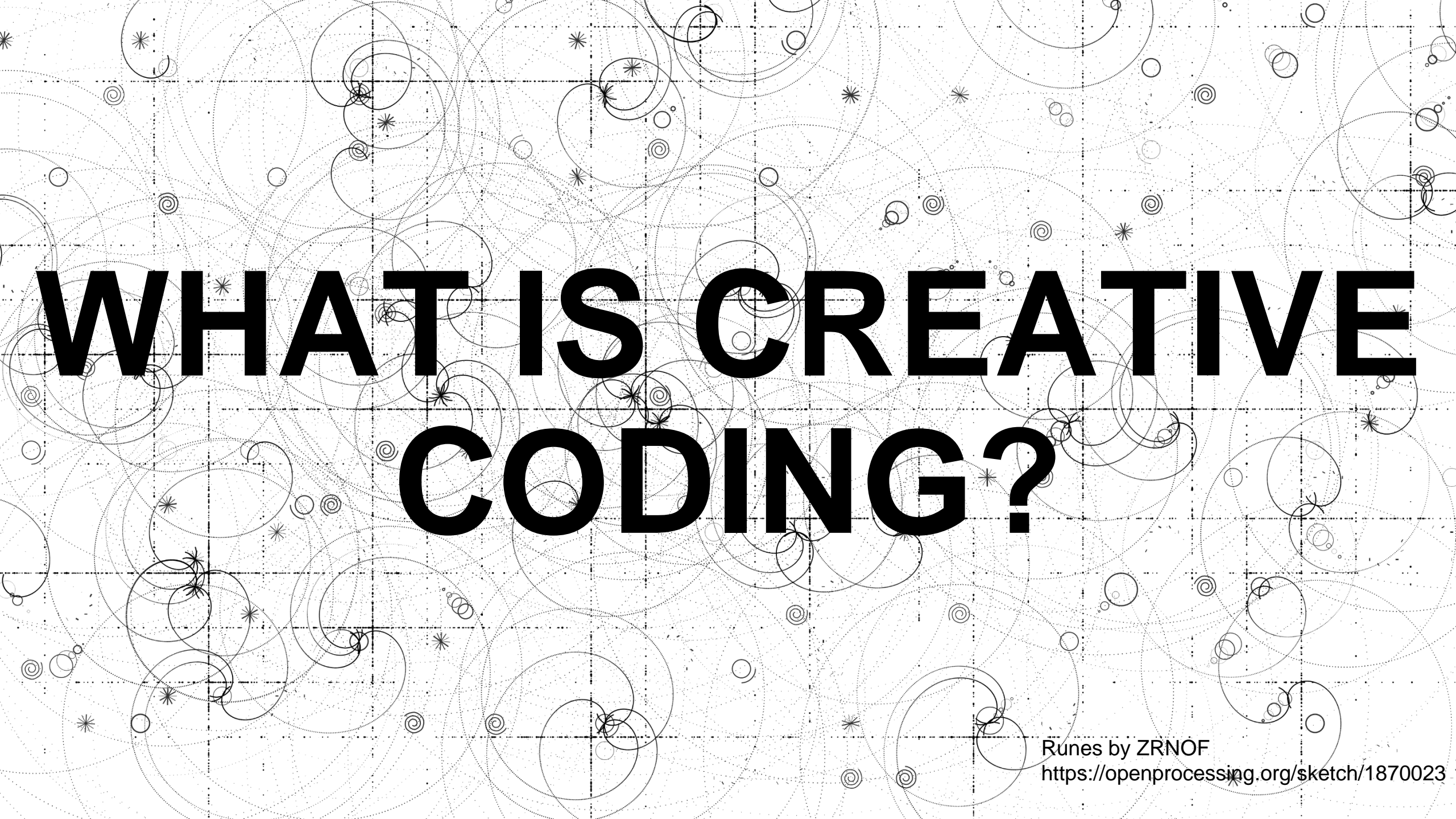


CREATIVE CODING

with p5.js

Fluid GPU-10 by Vamoss
<https://openprocessing.org/sketch/1799766>

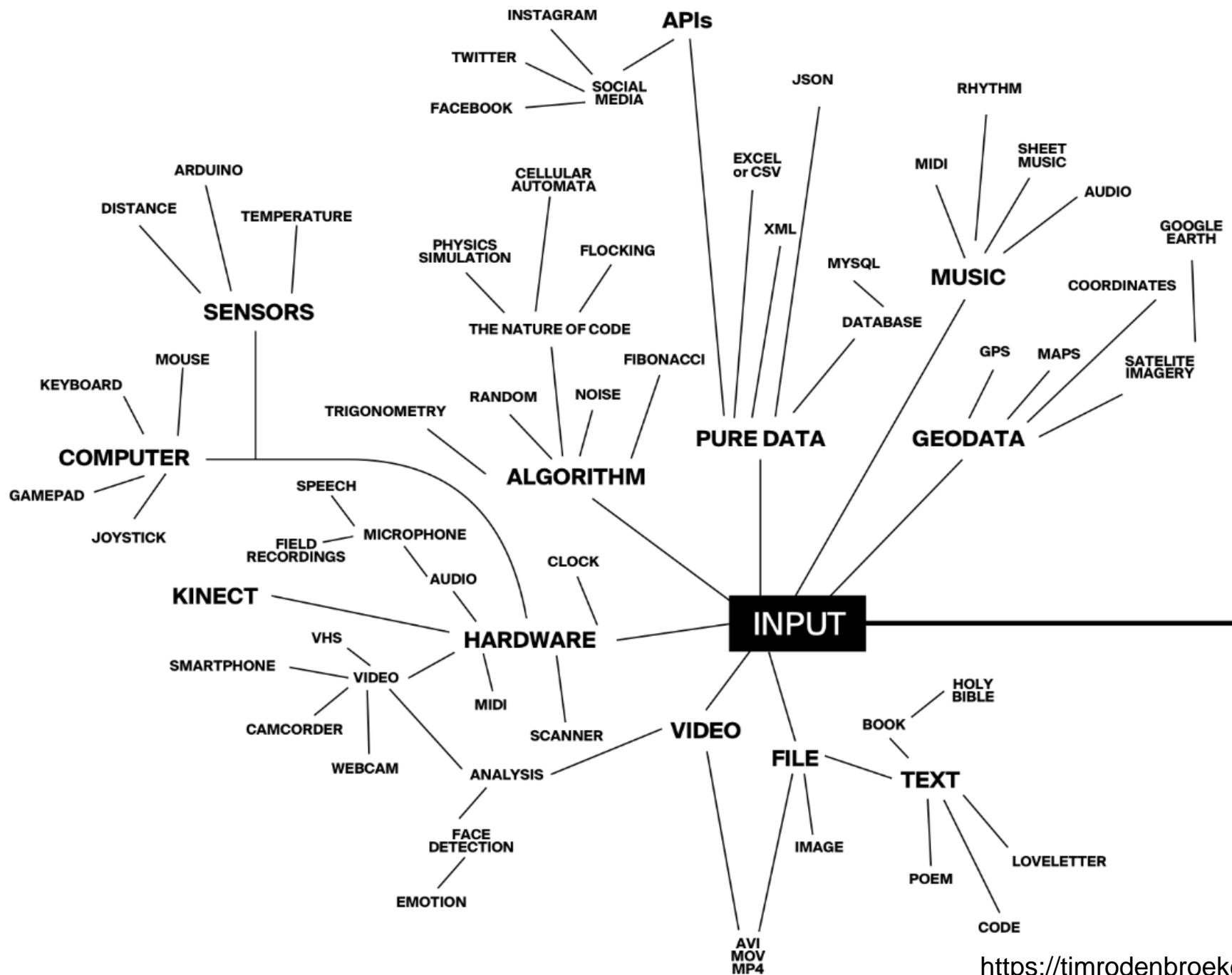


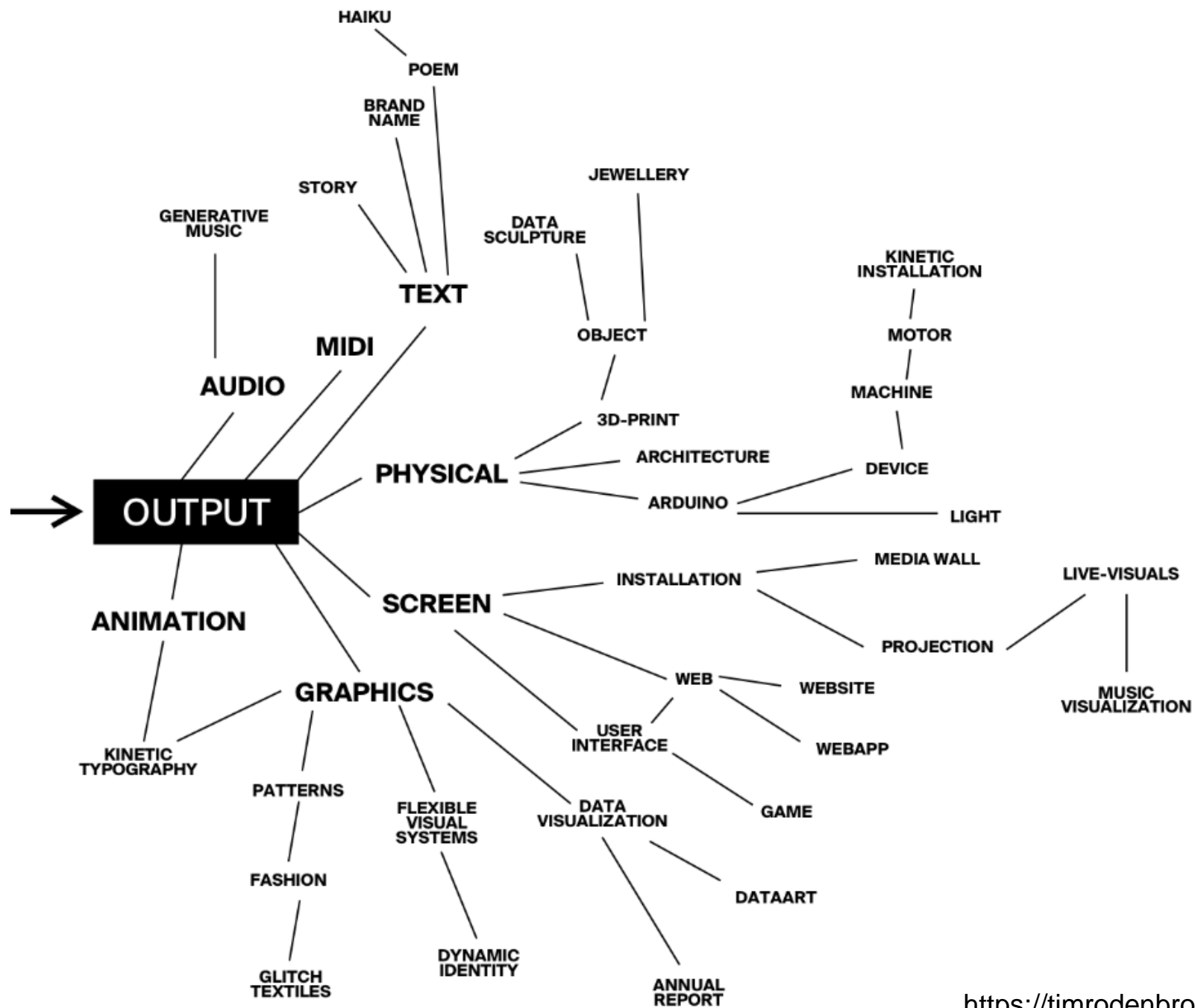
WHAT IS CREATIVE CODING?

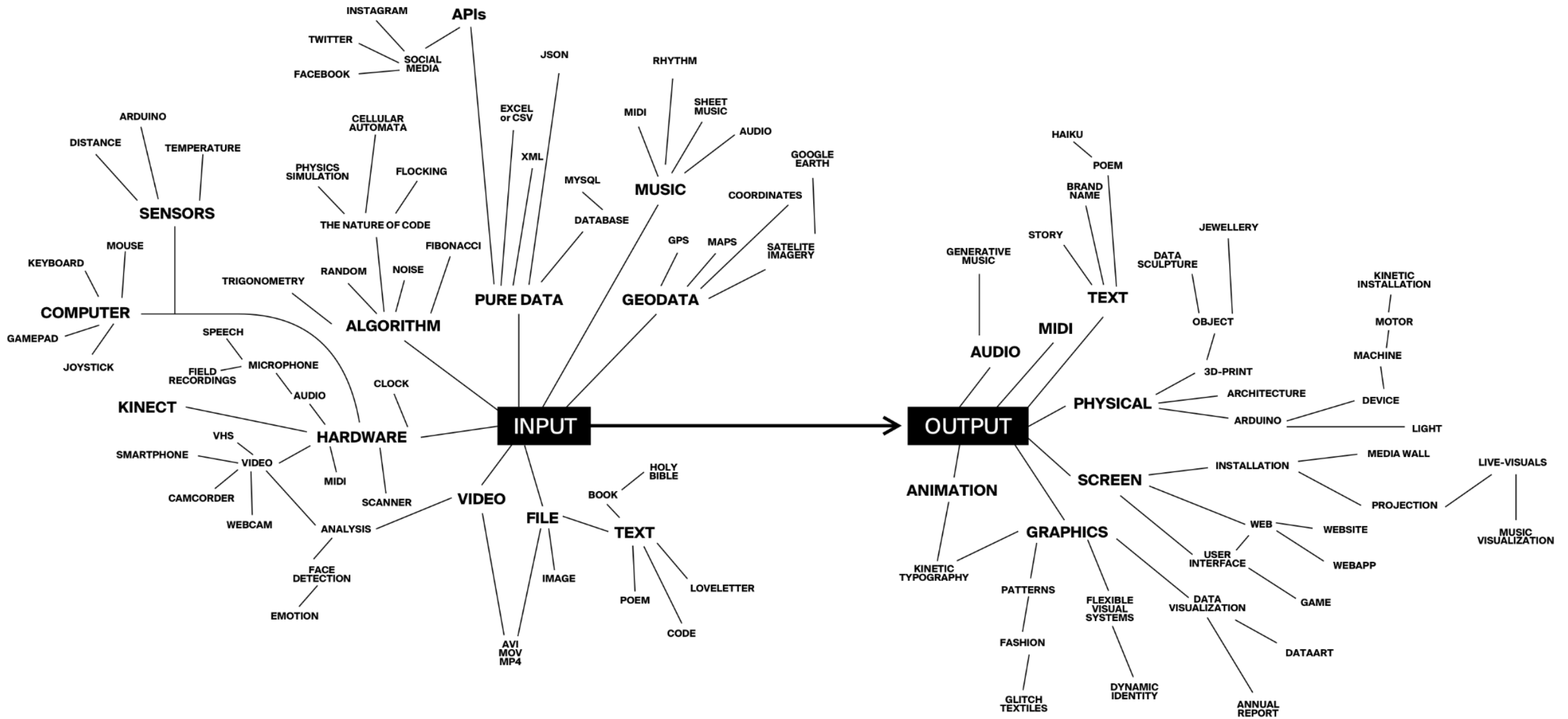
Runes by ZRNOF
<https://openprocessing.org/sketch/1870023>

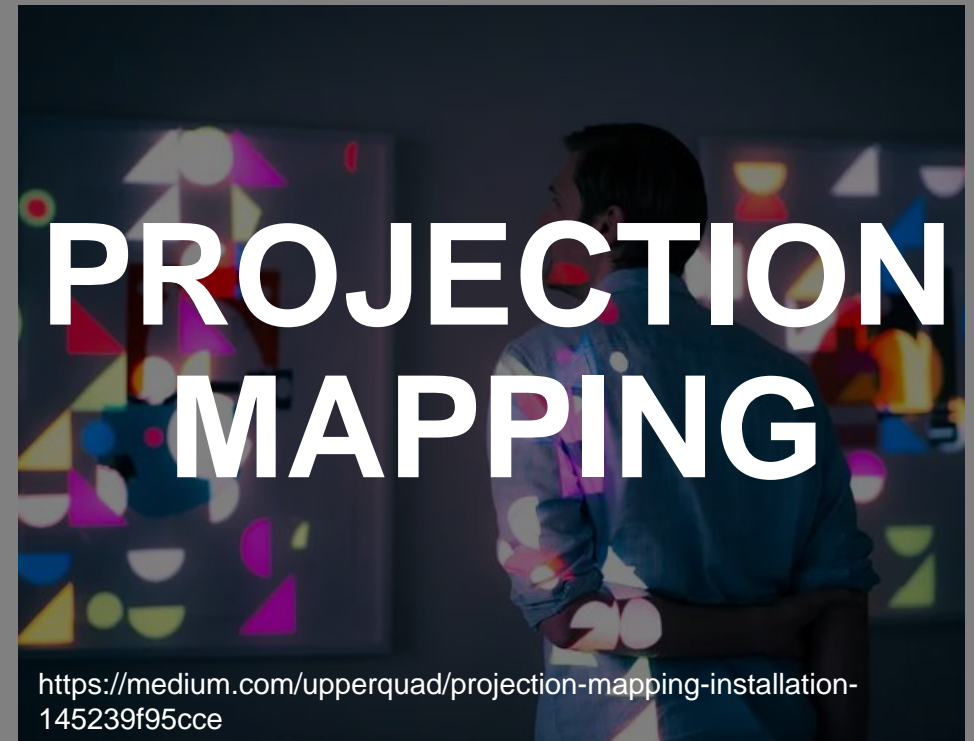
(Creative Coding) is a process, based on exploration, iteration, reflection and discovery, where code is used as the primary medium to create a wide range of media artifacts.

(Creative Coding) ist ein Prozess, der auf Erkundung, Iteration, Reflexion und Entdeckung beruht und bei dem Code als primäres Medium zur Schaffung einer breiten Palette von Medienartefakten eingesetzt wird.





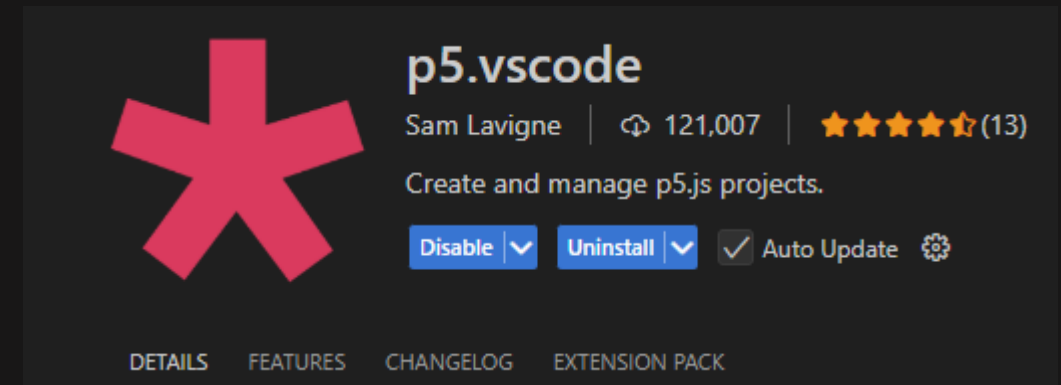




p5.js setup

Installation


- <https://p5js.org/tutorials/setting-up-your-environment/#vscode>
- VS Code extension: p5.vscode
- Easy setup possible with «Create p5.js Project»



Identifier samplavigne.p5-vscode

>|

Create p5.js Project

recently used 

First Steps

<https://editor.p5js.org/>

setup()

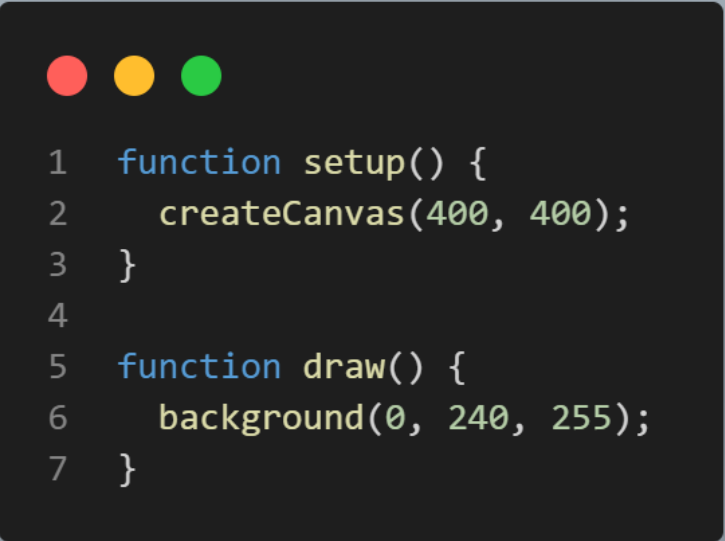
createCanvas(width, height)

draw()

background(v1, v2, v3)

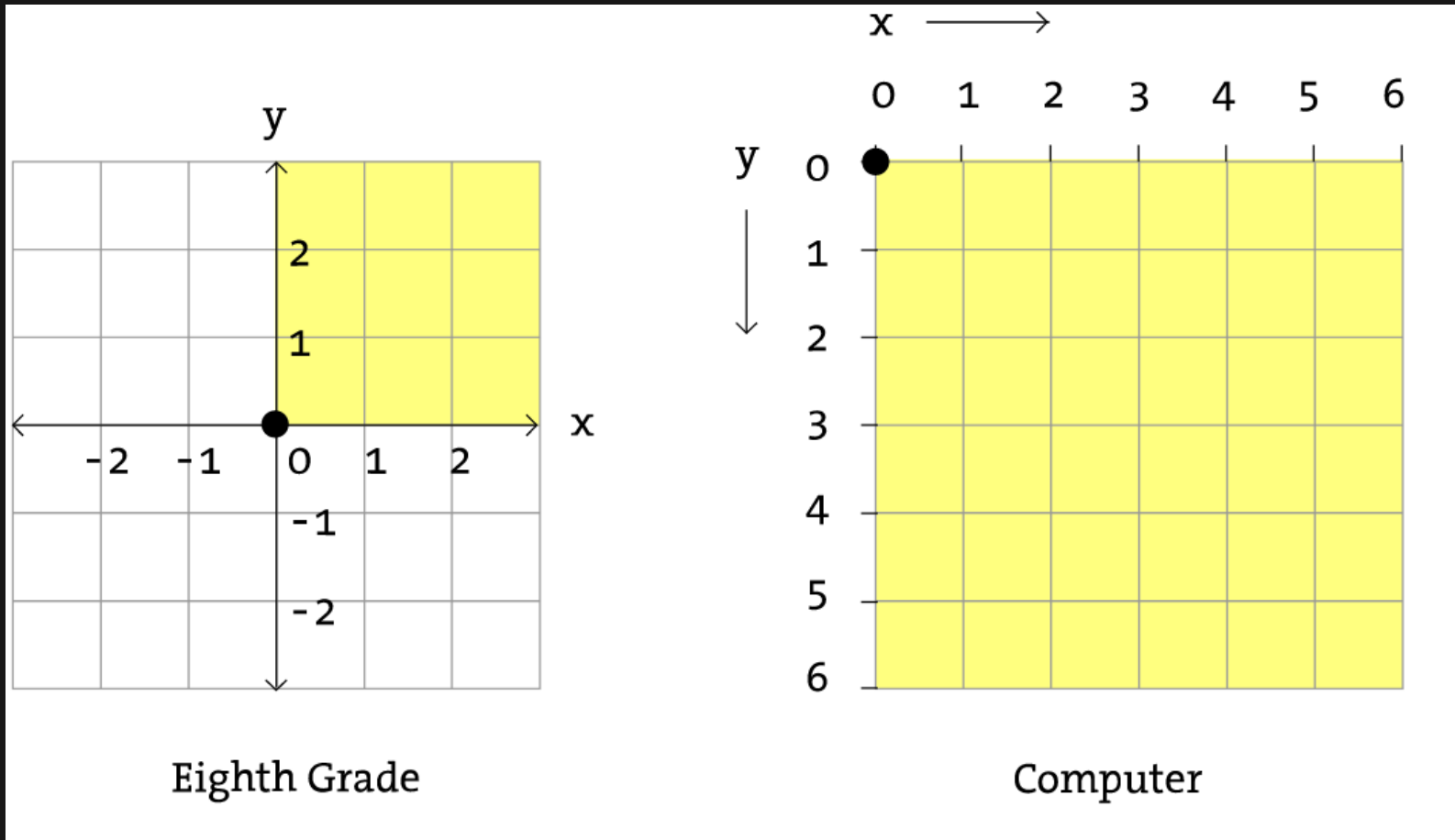
See reference for full
documentation:

<https://p5js.org/reference/>



```
1  function setup() {  
2      createCanvas(400, 400);  
3  }  
4  
5  function draw() {  
6      background(0, 240, 255);  
7  }
```

Coordinate System



Shapes & Stroke

stroke(color)

strokeWeight(weight)

point(x , y)

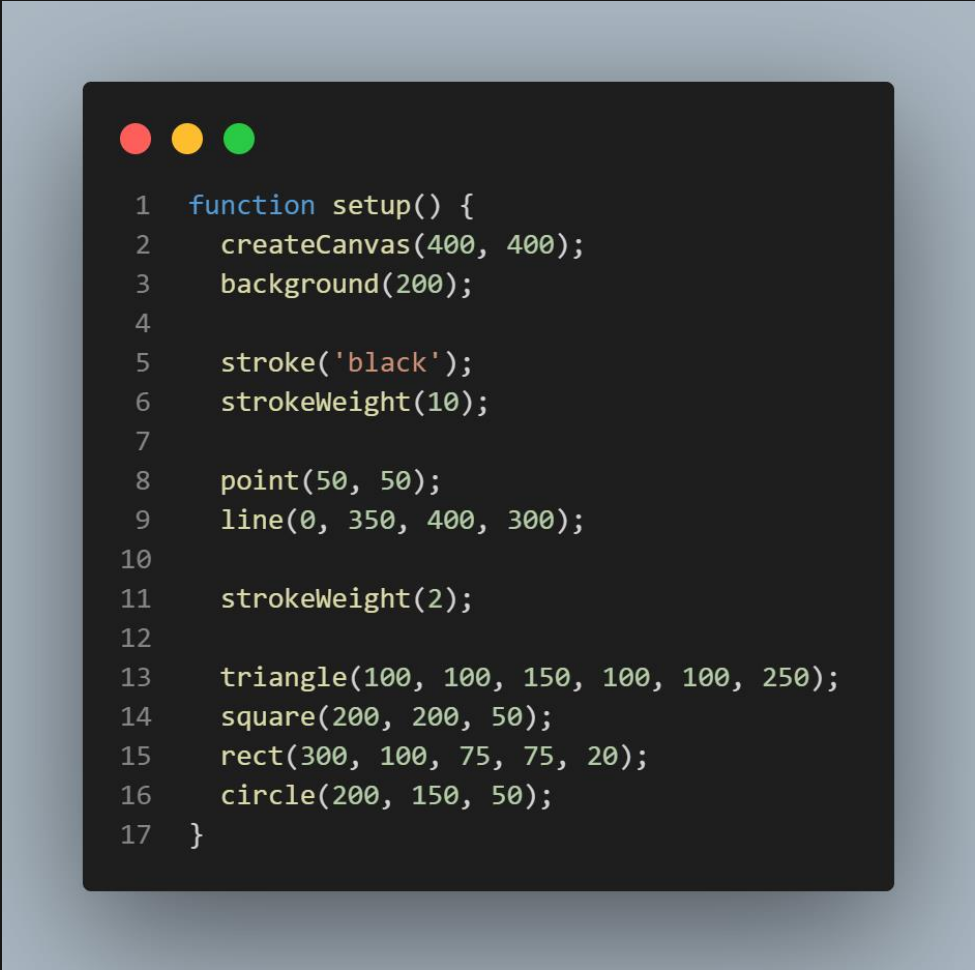
line(x1, y1, x2, y2)

triangle(x1, y1, x2, y2, x3, y3)

square(x, y, s)

rect(x, y, w, h, r)

circle(x, y, r)

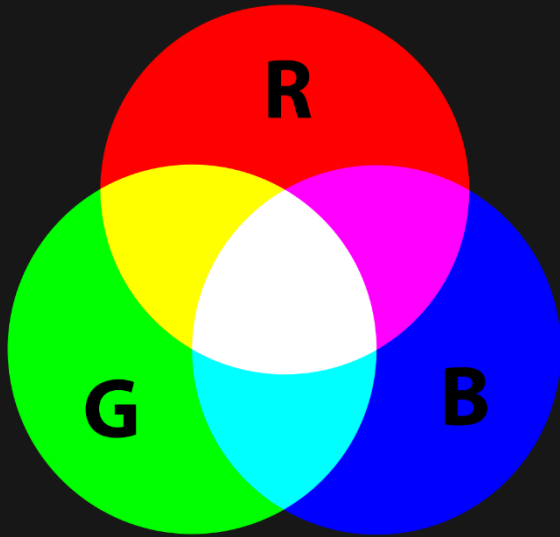


```
1  function setup() {  
2    createCanvas(400, 400);  
3    background(200);  
4  
5    stroke('black');  
6    strokeWeight(10);  
7  
8    point(50, 50);  
9    line(0, 350, 400, 300);  
10  
11   strokeWeight(2);  
12  
13   triangle(100, 100, 150, 100, 100, 250);  
14   square(200, 200, 50);  
15   rect(300, 100, 75, 75, 20);  
16   circle(200, 150, 50);  
17 }
```

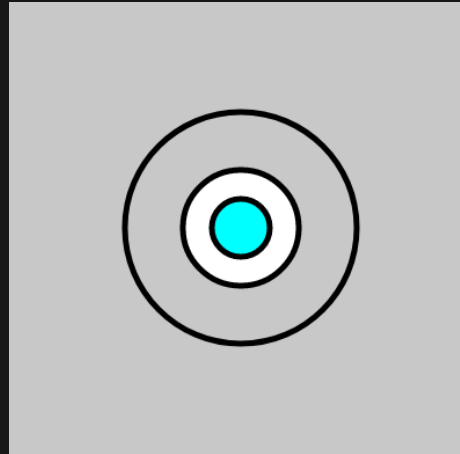
Color & (no)Fill

noFill()

fill(r, g, b)



<https://en.wiktionary.org/wiki/RGB>



```
1 function setup() {  
2   createCanvas(400, 400);  
3   background(200);  
4   stroke('black');  
5   strokeWeight(5);  
6  
7   noFill();  
8   circle(200, 200, 200);  
9  
10  fill(255, 255, 255);  
11  circle(200, 200, 100);  
12  
13  fill(0, 255, 255);  
14  circle(200, 200, 50);  
15 }
```

Variables (not all)

windowWidth
windowHeight

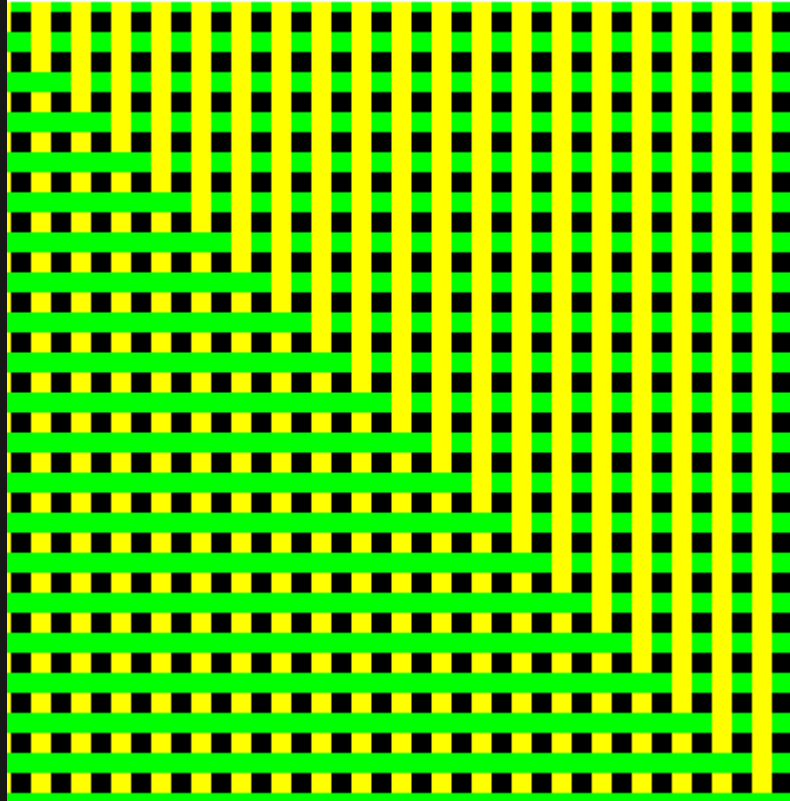
width
height

print()
console.log()



```
1  function setup() {  
2      createCanvas(windowWidth, windowHeight);  
3      background(100);  
4      square(width/4, height/4, width/2);  
5  
6      print(width)  
7      console.log(height);  
8  }  
9
```


Pattern example



```
1 function setup() {  
2   createCanvas(400, 400);  
3   strokeWeight(10);  
4 }  
5  
6 function draw() {  
7   background(0);  
8   for (let i = 0; i <= innerWidth; i += 20) {  
9     stroke(0, 255, 0);  
10    line(0, i, width, i);  
11  
12    stroke(255, 255, 0);  
13    line(i, 0, i, height);  
14  }  
15 }
```

Random

random([min], [max])

random(choises)



```
1 function setup() {  
2   createCanvas(windowWidth, windowHeight);  
3   strokeWeight(2);  
4 }  
5  
6 function draw() {  
7   background(0, 0, 0, 10);  
8   for(let i = 0 ;i < 10; i++) {  
9     stroke(random(255), random(255), random(255));  
10    let x = random(0, width);  
11    let y = random(0, height);  
12  
13    line(width / 2, height / 2, x, y);  
14  
15    // also try this:  
16    // line(x, y, y, x);  
17  }  
18 }  
19  
20
```

Interactivity

mouseX

mouseY

mousePressed()

keyPressed()

More:

<https://archive.p5js.org/learn/interactivity.html> (Archived, but still valuable)

```
1  let r, g, b = 0;
2
3  function setup() {
4    createCanvas(400, 400);
5    strokeWeight(10);
6  }
7
8  function draw() {
9    background(0);
10   stroke(r, g, b);
11   fill(r, g, b, 128);
12   ellipse(width/2, height/2, mouseX, mouseY);
13 }
14
15 function mousePressed() {
16   r = random(255);
17   g = random(255);
18   b = random(255);
19 }
20
```


Multiple Canvases

- <https://p5js.org/examples/advanced-canvas-rendering-multiple-canvases/>

```
index.html ×
sketches > 06 > index.html > html > body > script
Jooky2, 3 months ago | 1 author (Jooky2)
1 <!DOCTYPE html>
2 <html lang="de">
3   <head>
4     <title>Sketch 6</title>
5     <meta charset="UTF-8" />
6     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7     <link rel="stylesheet" type="text/css" href="../../css/base.css" />
8   </head>
9   <body>
10    <script src="../../js/p5.min.js"></script>
11    <script src="../../sketch1.js"></script>
12    <script src="../../sketch2.js"></script>
13  </body>
14 </html>
15

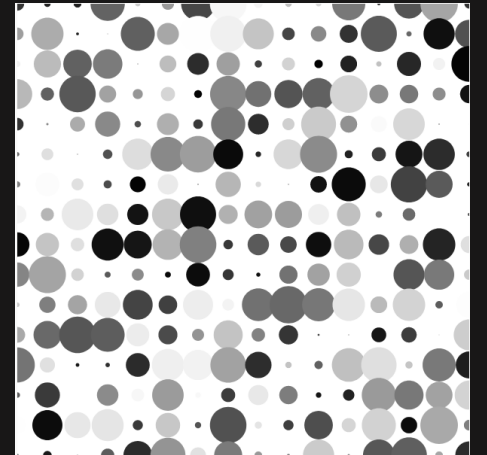
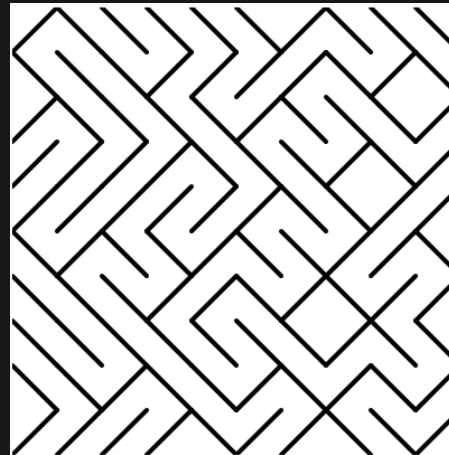
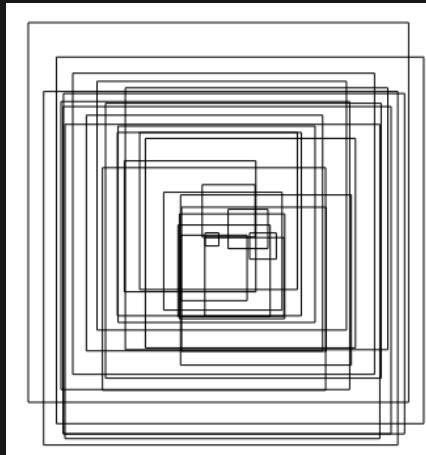
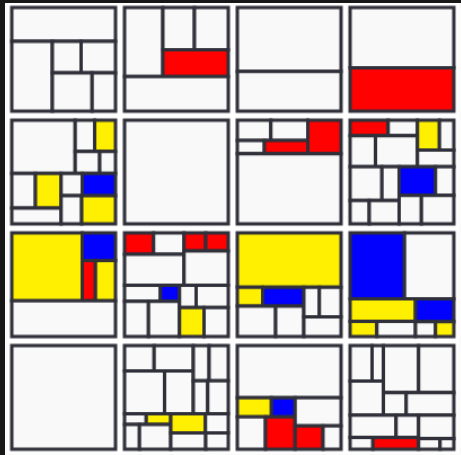
sketch1.js ×
sketches > 06 > sketch1.js > ...
Jooky2, 3 months ago | 1 author (Jooky2)
1 function sketch1(p) {
2   p.setup = function () {
3     p.createCanvas(720, 200);
4     p.background(0);
5   };
6   p.draw = function () {
7     p.circle(p.mouseX, p.mouseY, 50);
8   };
9 }
10
11 new p5(sketch1);

sketch2.js M ×
sketches > 06 > sketch2.js > ...
You, 2 seconds ago | 2 authors (Jooky2 and one other)
1 function sketch2(p) {
2   p.setup = function () {
3     p.createCanvas(720, 200);
4     p.background(255);
5     p.fill(0);
6     p.stroke(255);
7   };
8   p.draw = function () {
9     p.square(p.mouseX, p.mouseY, 50);
10  };
11 }
12
13 new p5(sketch2);
14

<main>
  <canvas id="defaultCanvas0" class="p5Canvas" style="width: 720px; height: 200px;" width="720" height="200"></canvas>
  <canvas id="defaultCanvas1" class="p5Canvas" style="width: 720px; height: 200px;" width="720" height="200"></canvas>
</main>
```

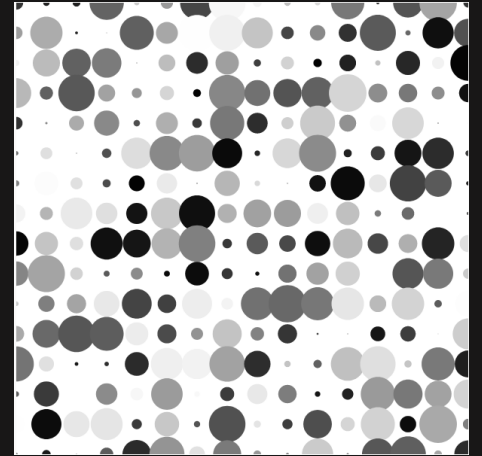
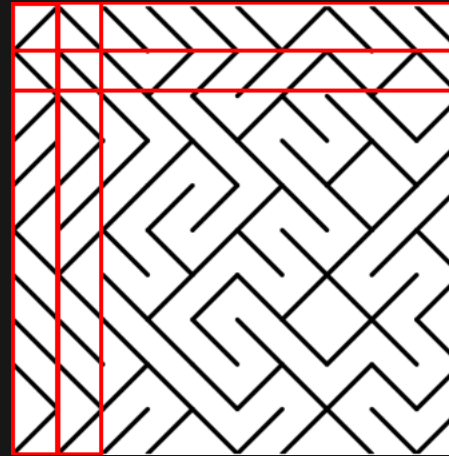
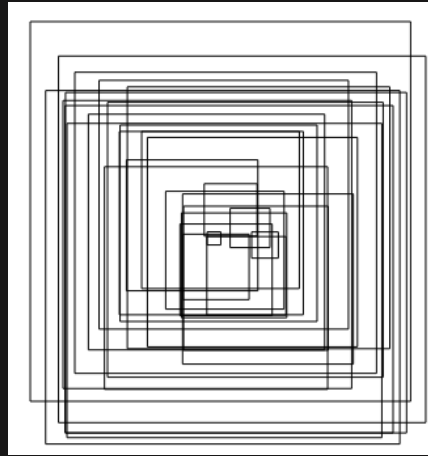
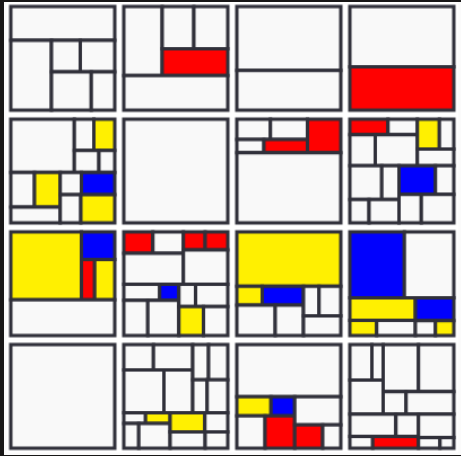
Tasks

1. Setup project with git (and maybe vue) on github (public!)
2. Send me the link to your repository
3. Recreate atleast two of the following:



3. Add interactivity to them
4. Create own sketches with what you have learned so far
5. Tips for the tasks above on the next slide (skip for a challenge)

Tips



1. First make a 4x4 grid of squares, then add «random» rectangles. This can be either truly random generated rectangles or from a list of predefined patterns.
2. Create random squares in the middle of your canvas
3. This one is probably the trickiest one to see, but it is just a grid of squares with a random diagonal line (only two possibilities)
4. This is a grid with random circle sizes and colors.