

# Kreativ programmering

Anders Larsen, utvikler

# Hva skal vi bruke?



## Mitt foretrukne verktøy

- JavaScript-bibliotek for kreativ koding
- Bygger videre på HTML Canvas
- Mange funksjoner og variabler
- Støtter både 2D og 3D
- Kjører i nettleseren!

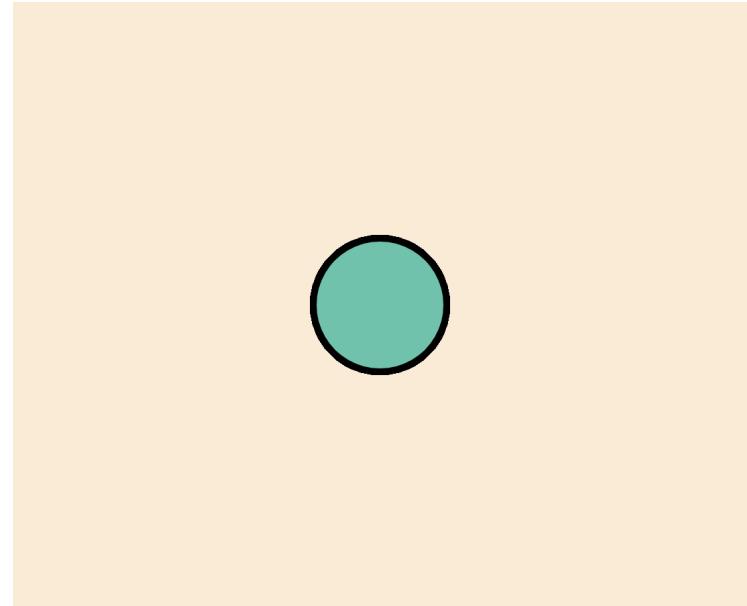
# Hvordan bruker man p5? setup() og draw()

```
// Kjøres én gang
function setup() {
    createCanvas(550, 454);

    fill("#70c2ad");
    strokeWeight(5);

    background("#FAEBD7");
    circle(width / 2, height / 2, 100);
}

// Kjøres flere ganger i sekundet
function draw() { }
```

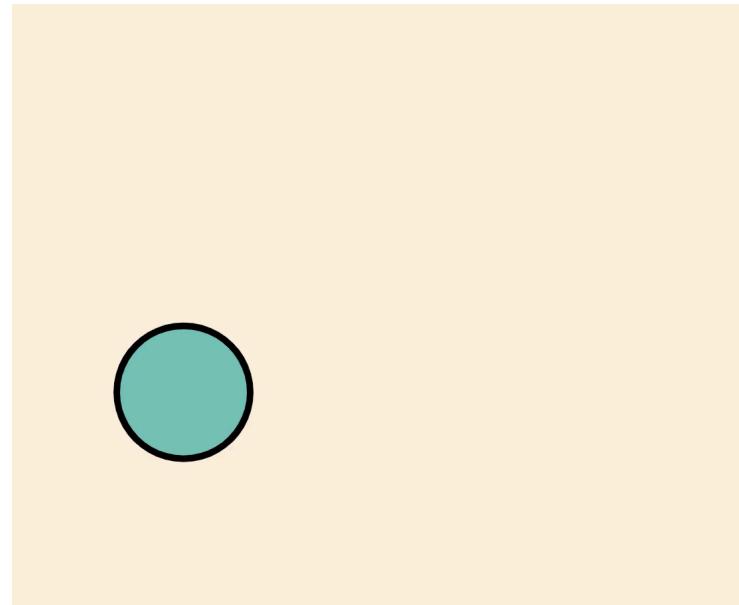


# Hvordan bruker man p5? setup() og draw()

```
// Kjøres én gang
function setup() {
    createCanvas(550, 454);

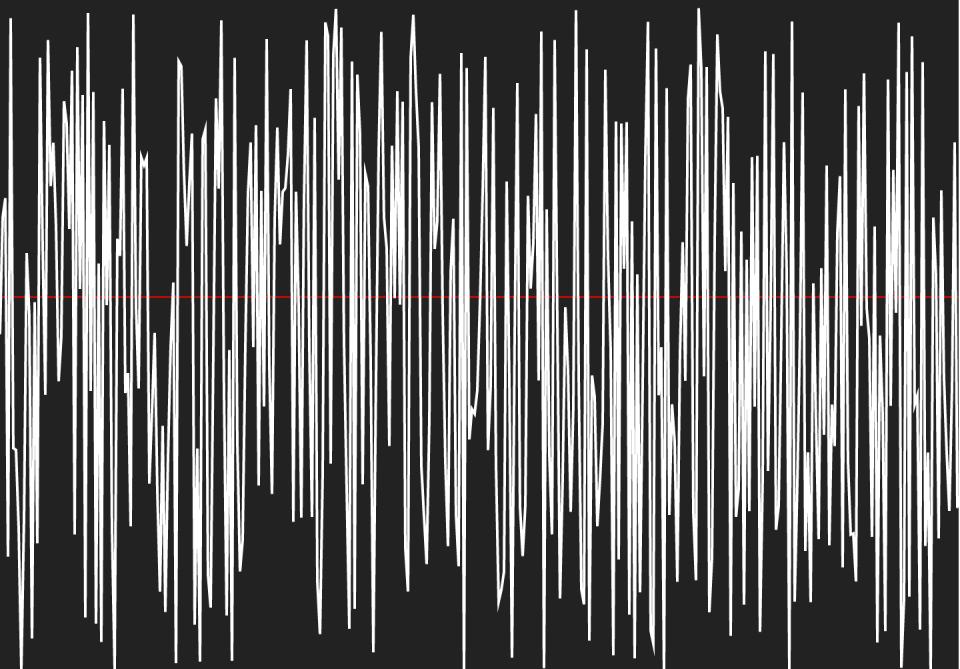
    fill("#70c2ad");
    strokeWeight(5);
}

// Kjøres flere ganger i sekundet
function draw() {
    background("#FAEBD7");
    circle(random(width), random(height), 100);
}
```

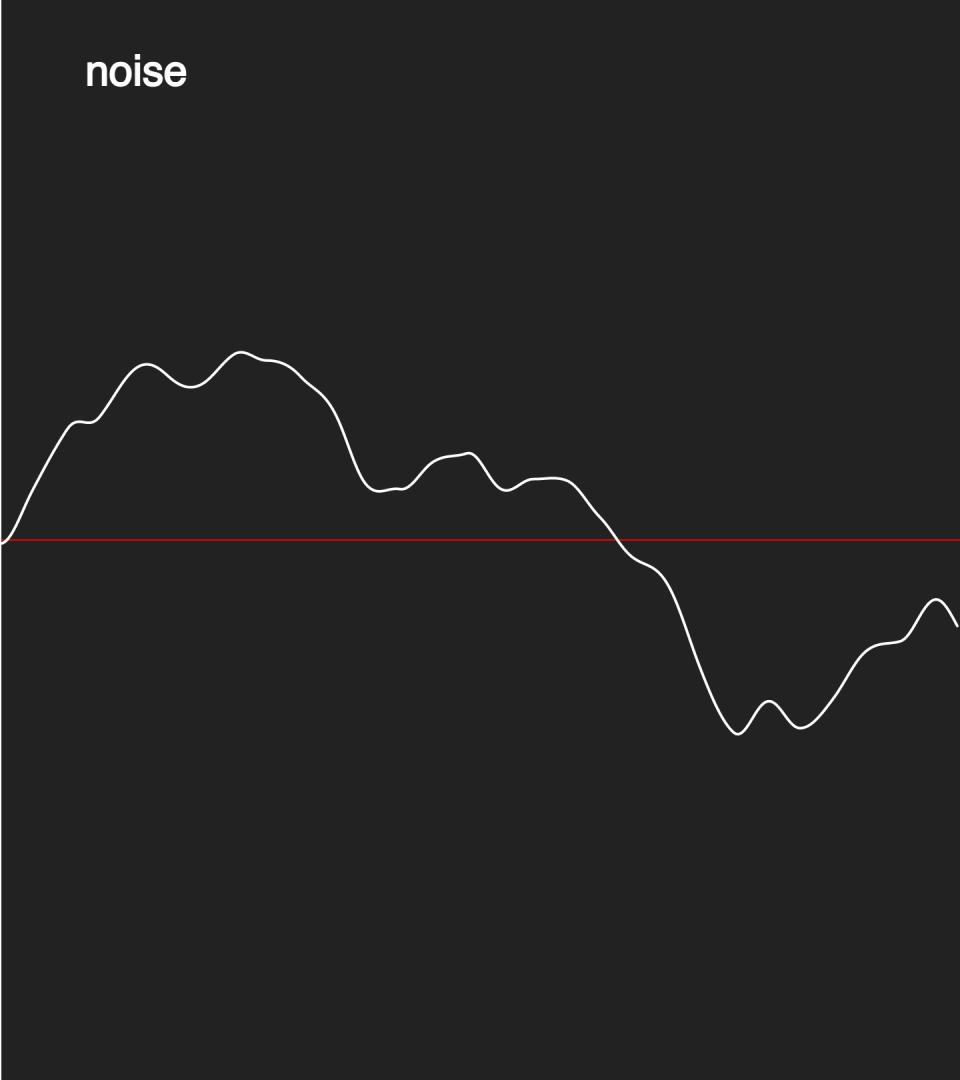


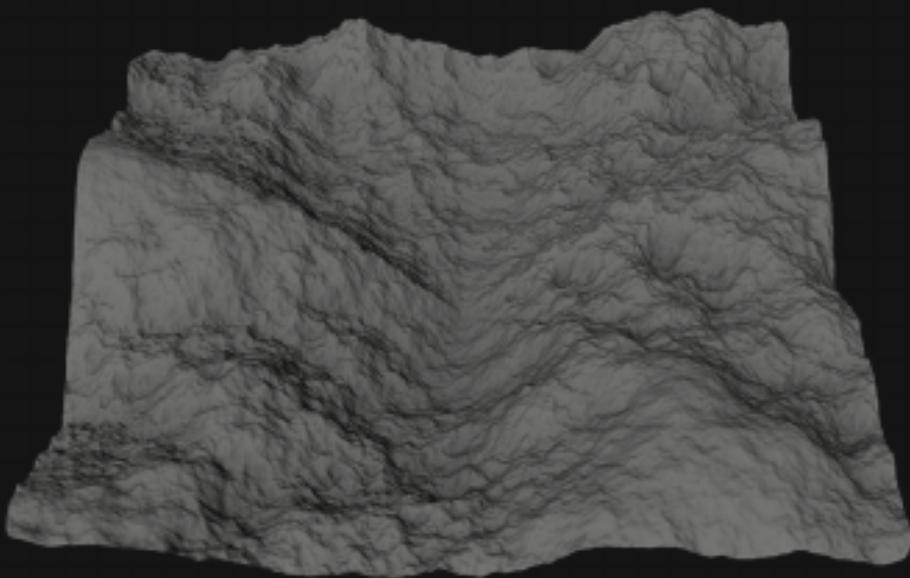
Kulere tilfeldighet, også kjent  
som støy

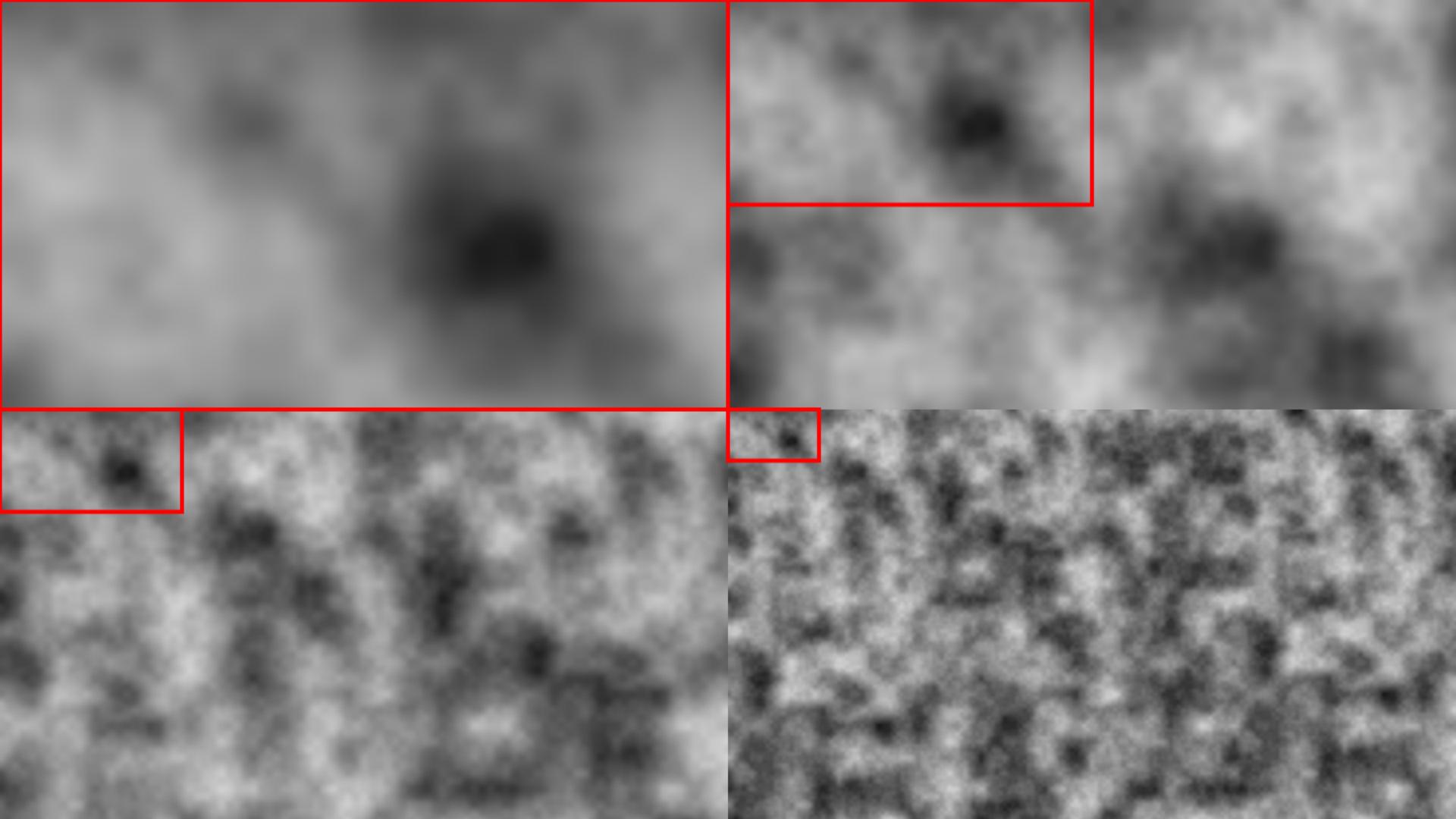
random



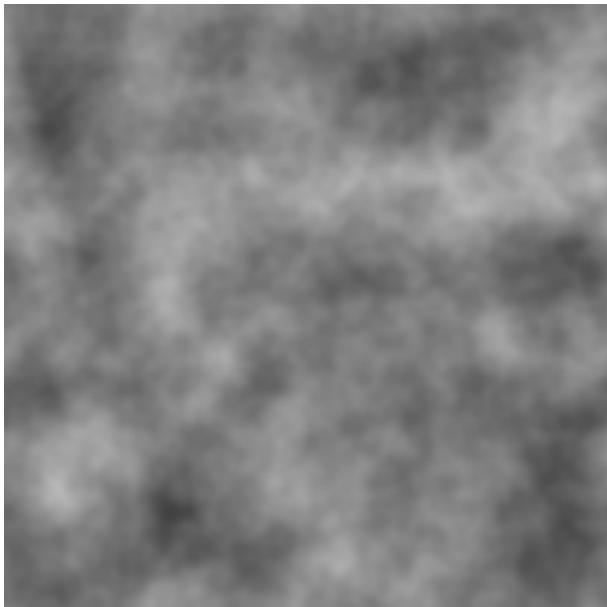
noise







# Perlinstøy i praksis



`noise()`

- Returnerer et tall mellom 0 og 1
- Tar inn mellom 1 og 3 parametere, typisk da x-, y-, og z-koordinater
- Vi jobber i 2D, men den tredje parameteren kan brukes som tid

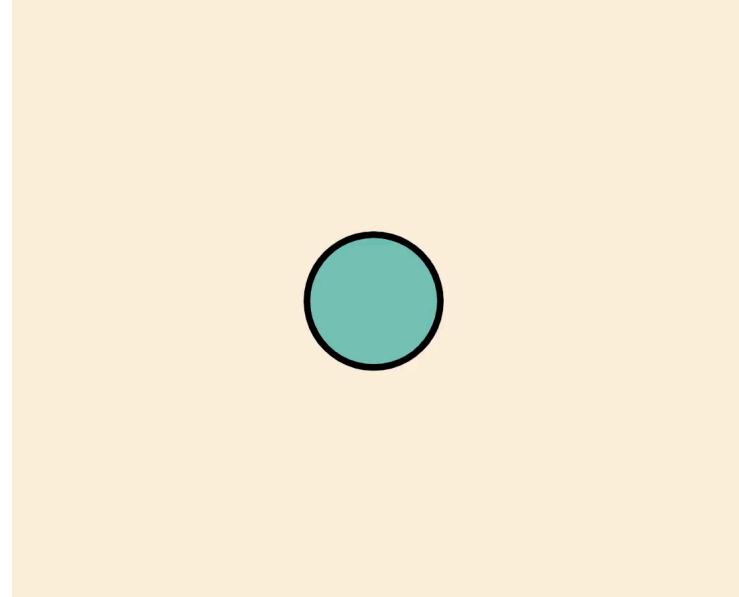
# La oss demonstrere støy

```
function setup() {
  createCanvas(550, 454);

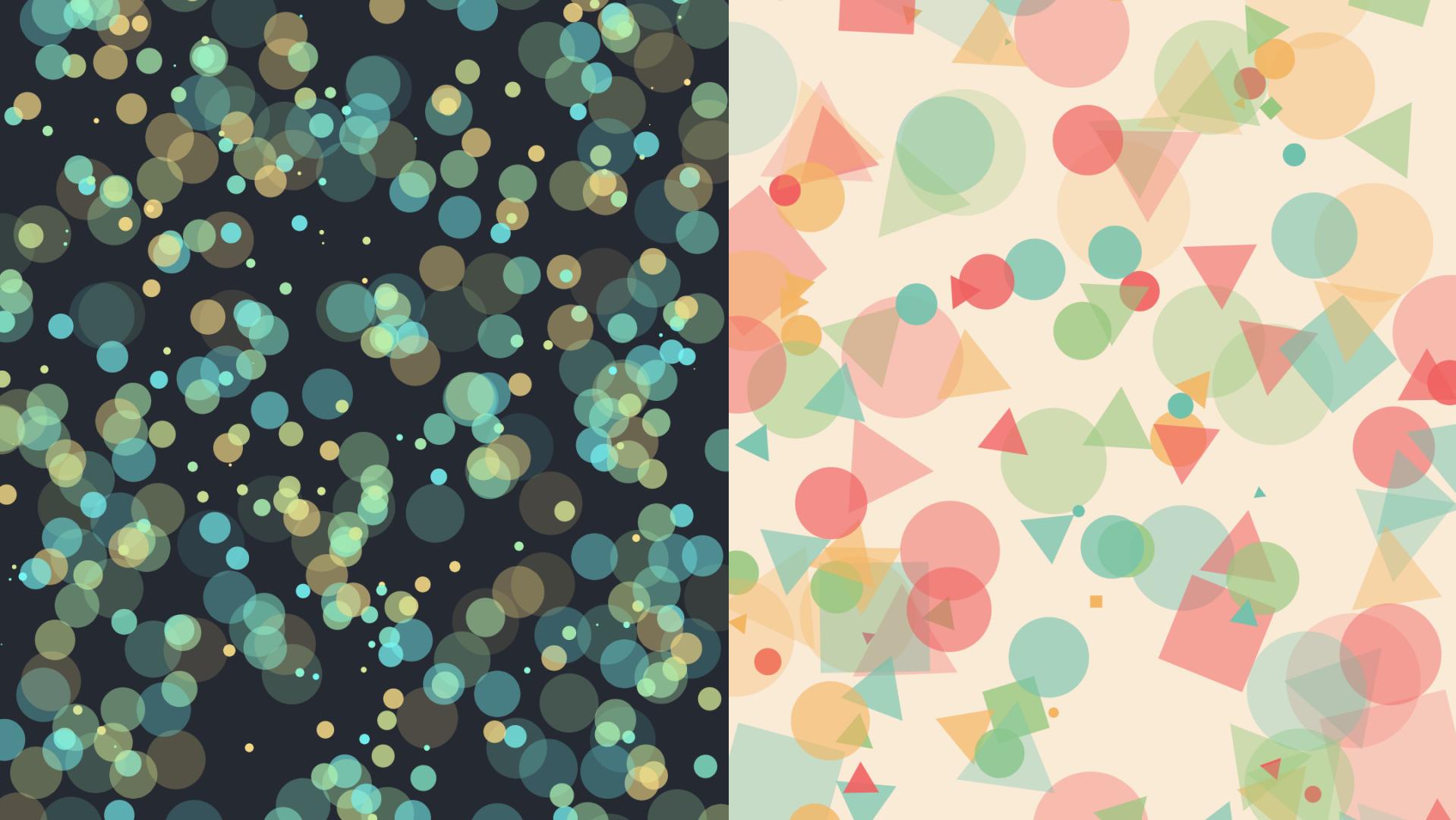
  fill("#70c2ad");
  strokeWeight(5);
  x = width / 2;
  y = height / 2;
}

function draw() {
  background("#FAEBD7");

  const a = 0.5 - noise(
    x * 0.005,
    y * 0.005,
    frameCount * 0.005
  );
  x += 4 * cos(2 * a * TWO_PI);
  y += 4 * sin(2 * a * TWO_PI);
  circle(x, y, 100);
}
```

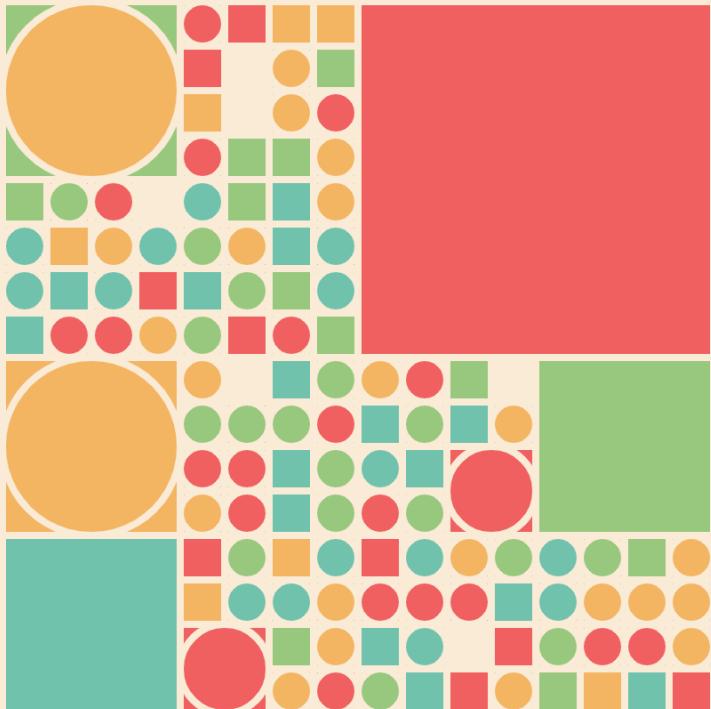
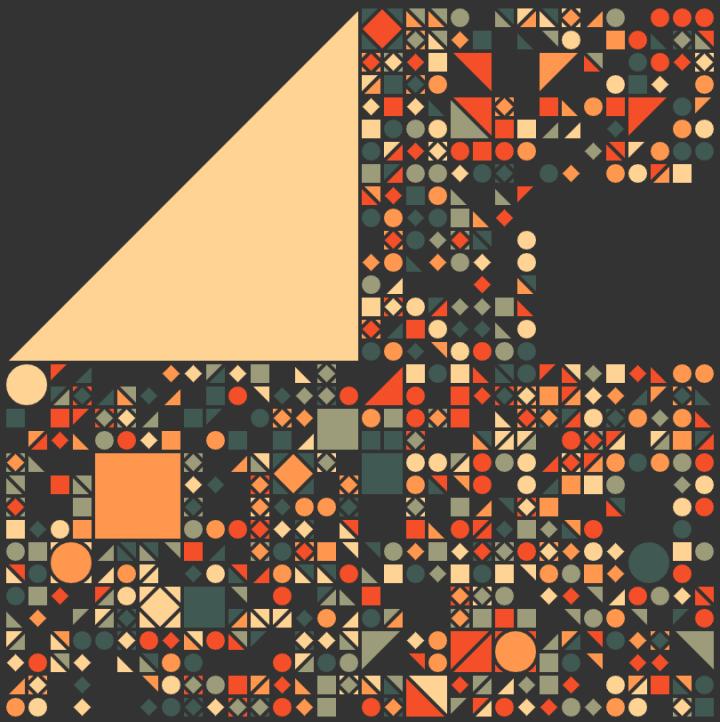


Tilfeldig plassering av former  
bokeh.js

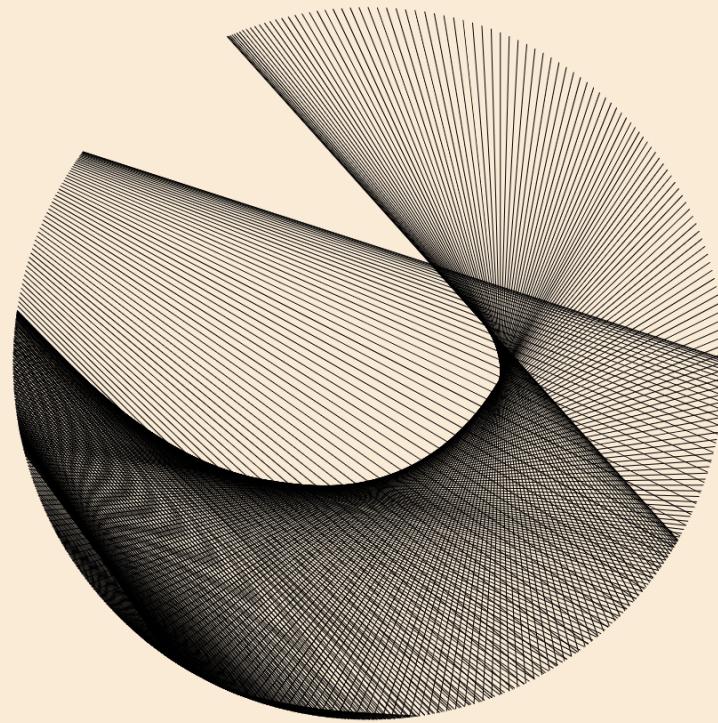
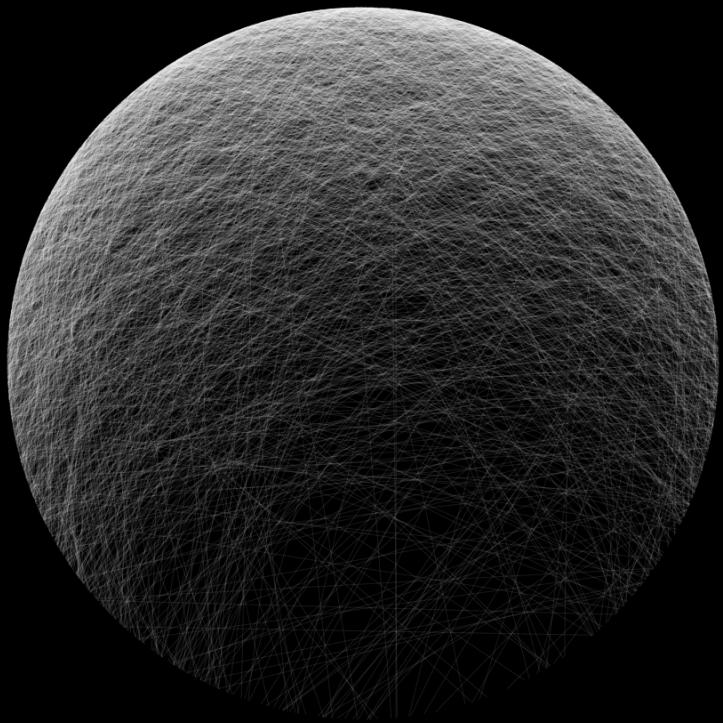


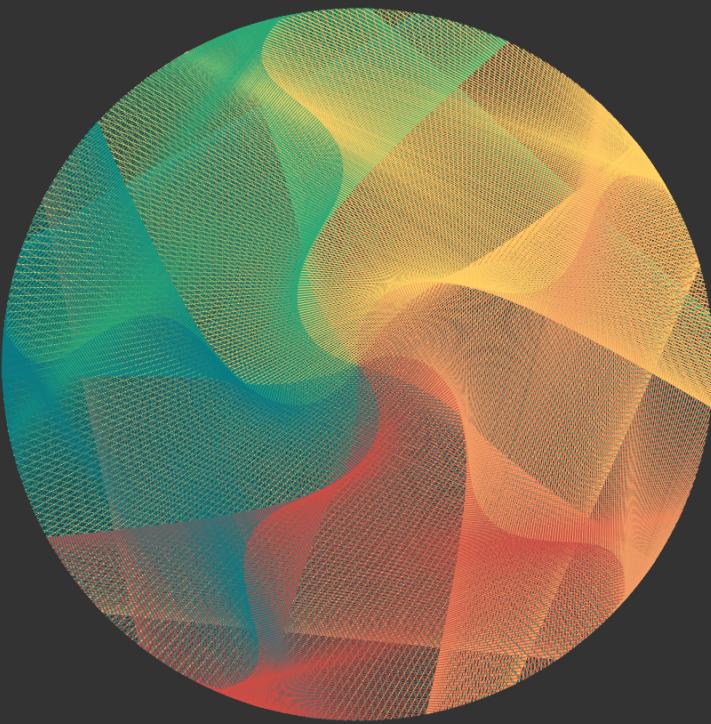
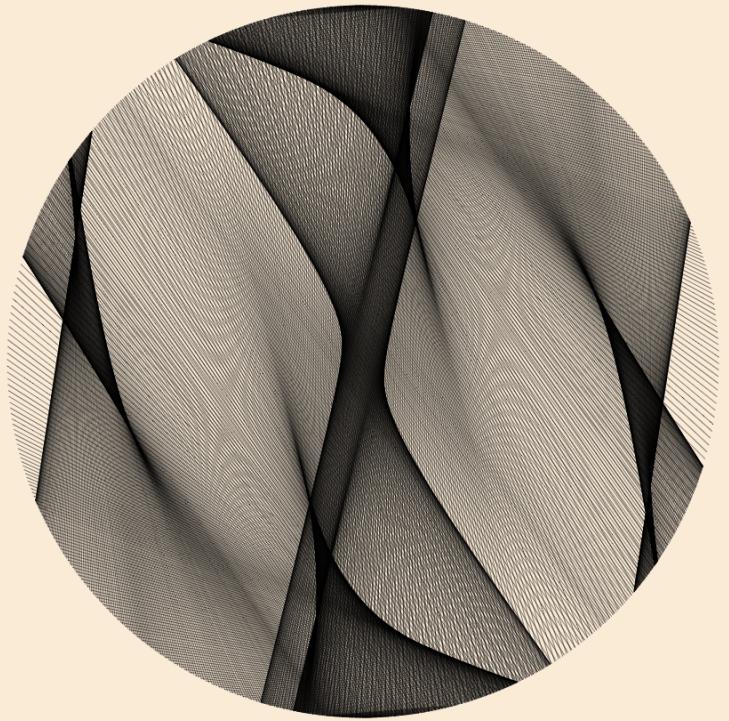
# Rekursjon med tilfeldighet

## recursion.js

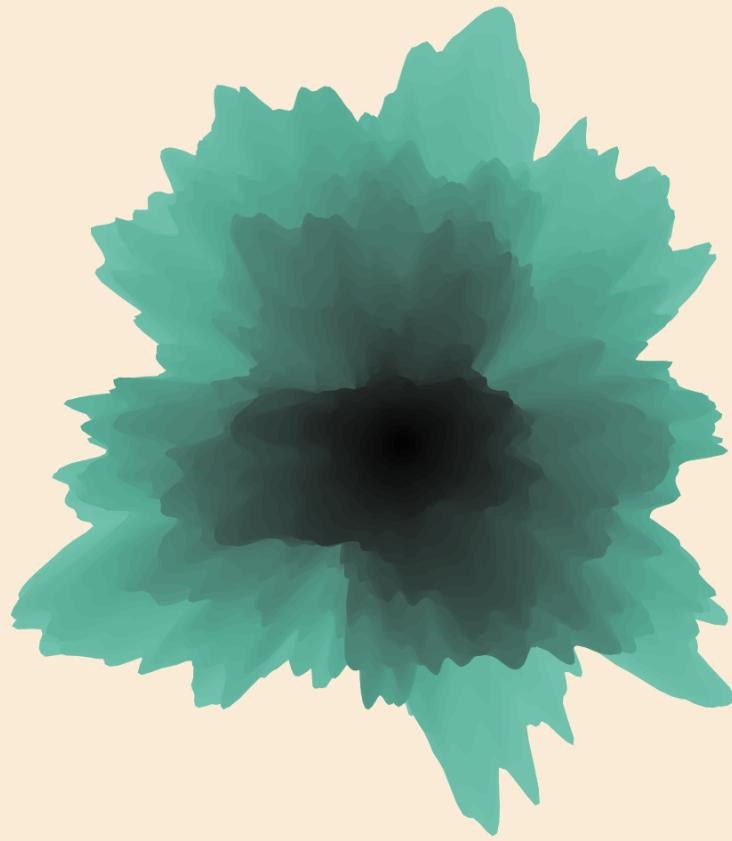
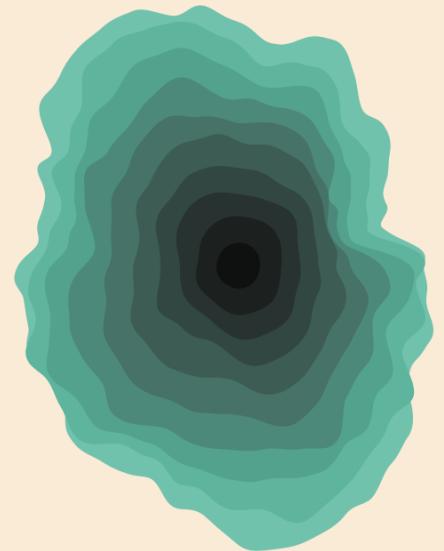


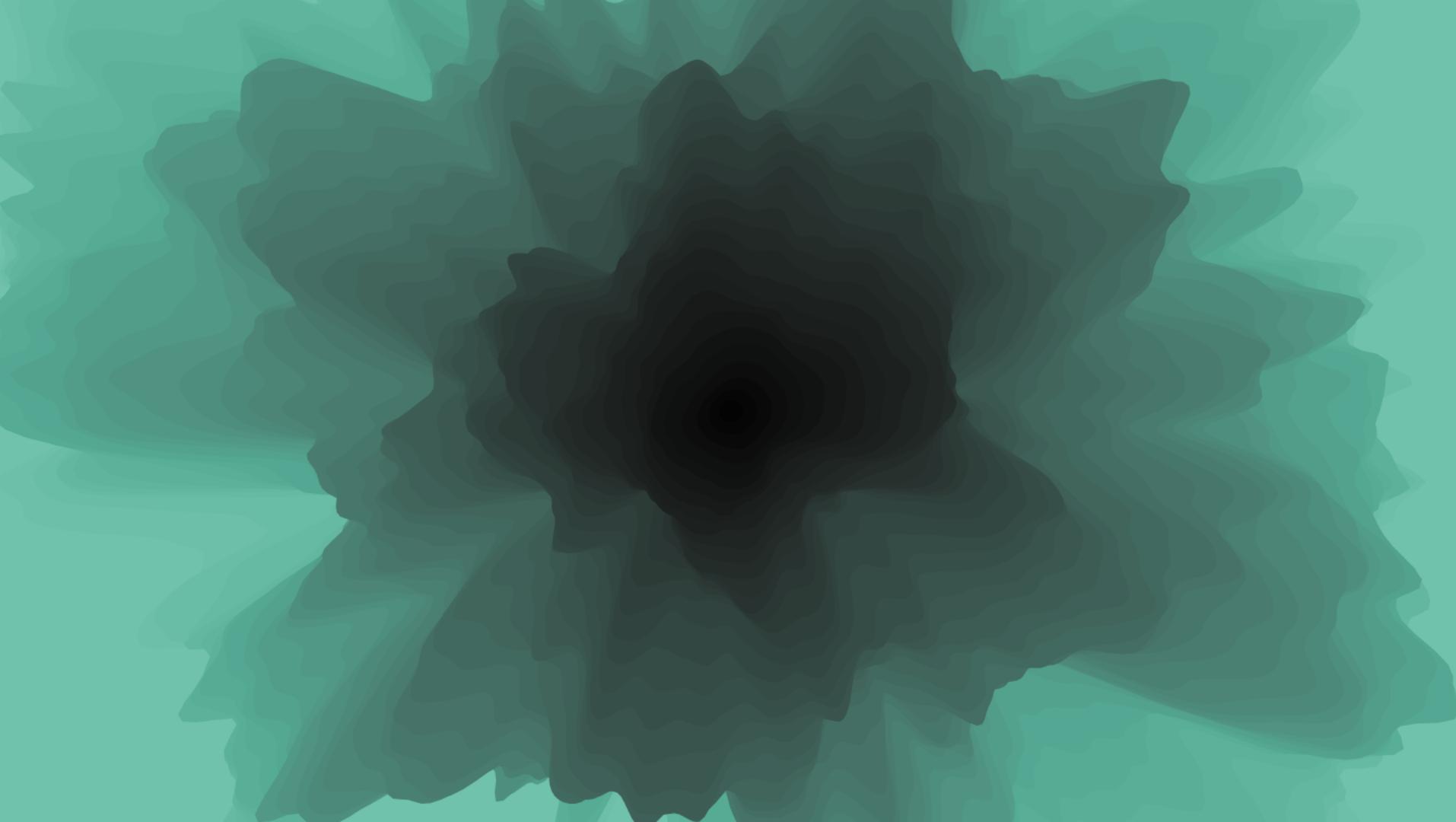
Linjer mellom to punkter  
chords.js

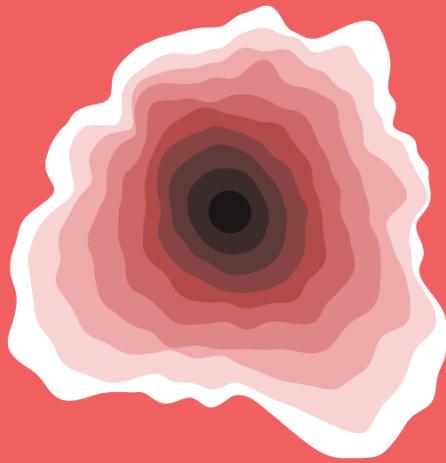




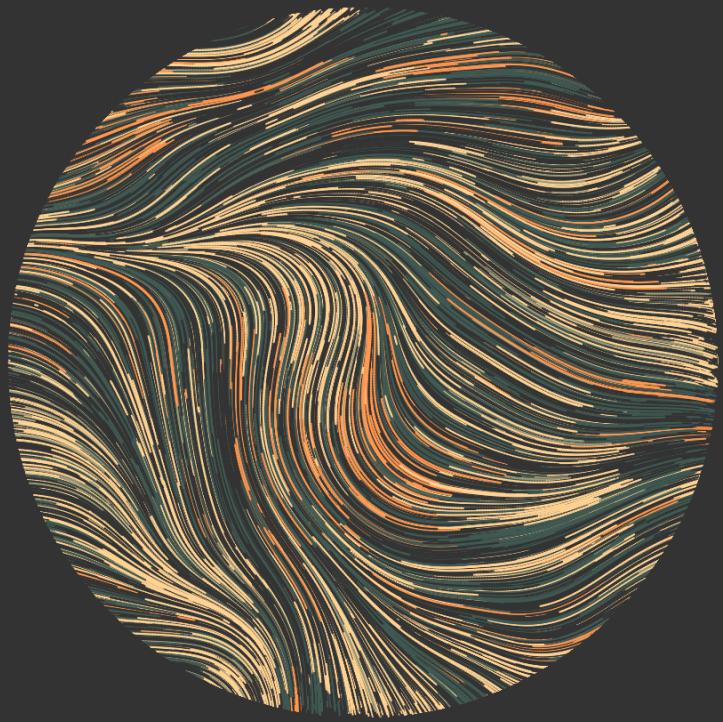
Blobbete sirkler med flere lag  
layers.js

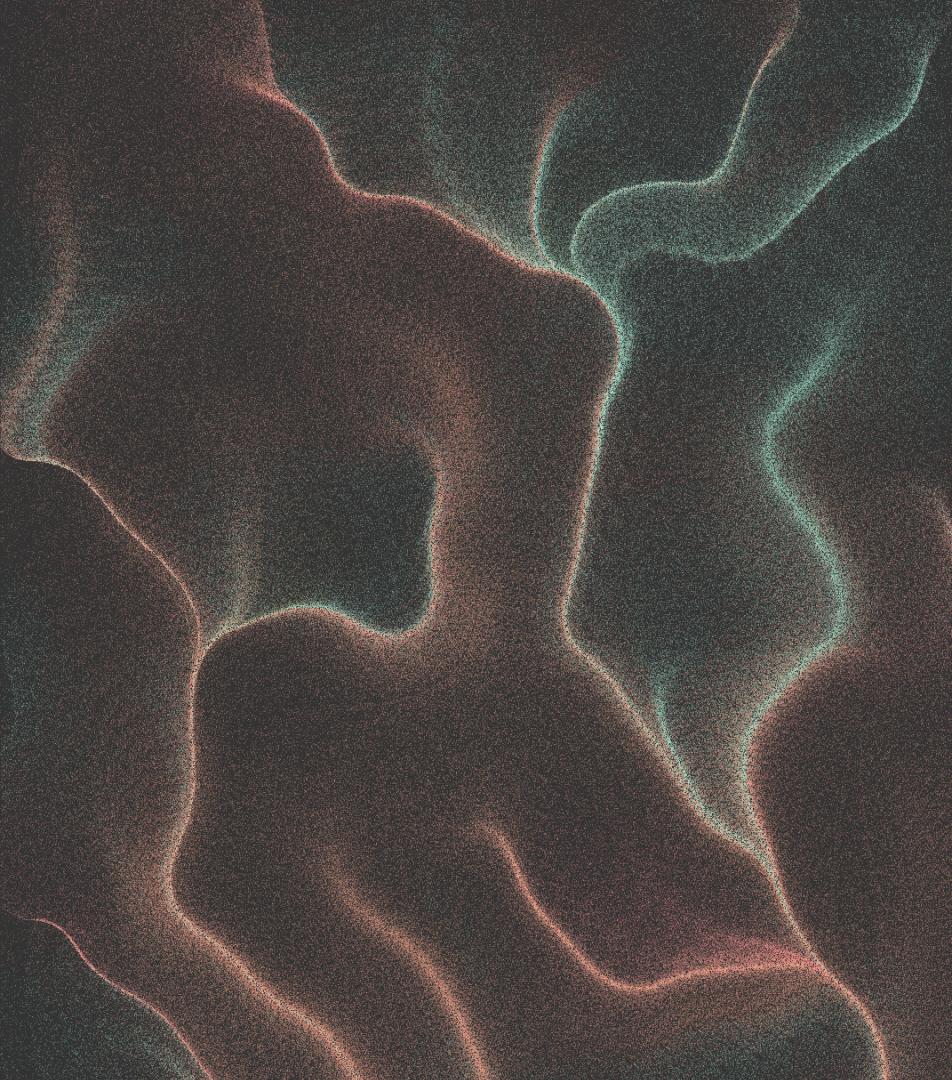


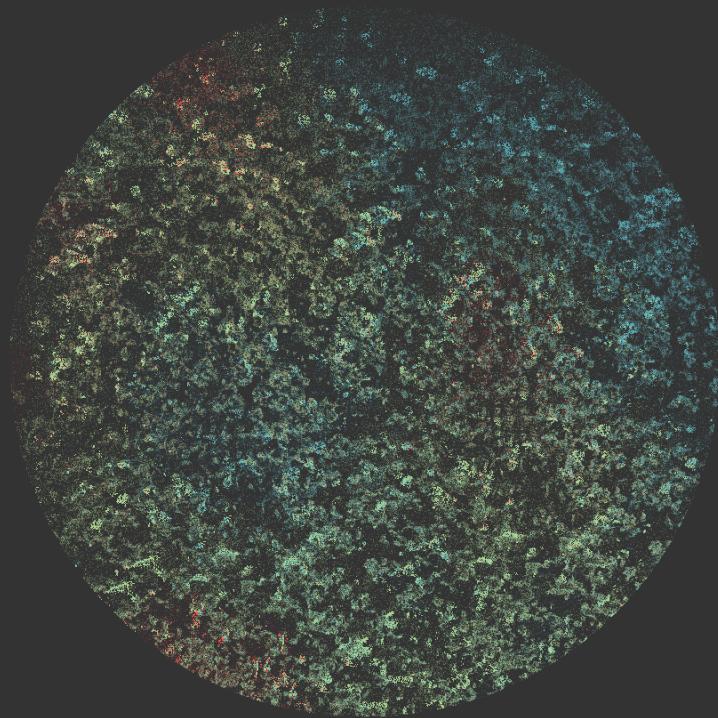
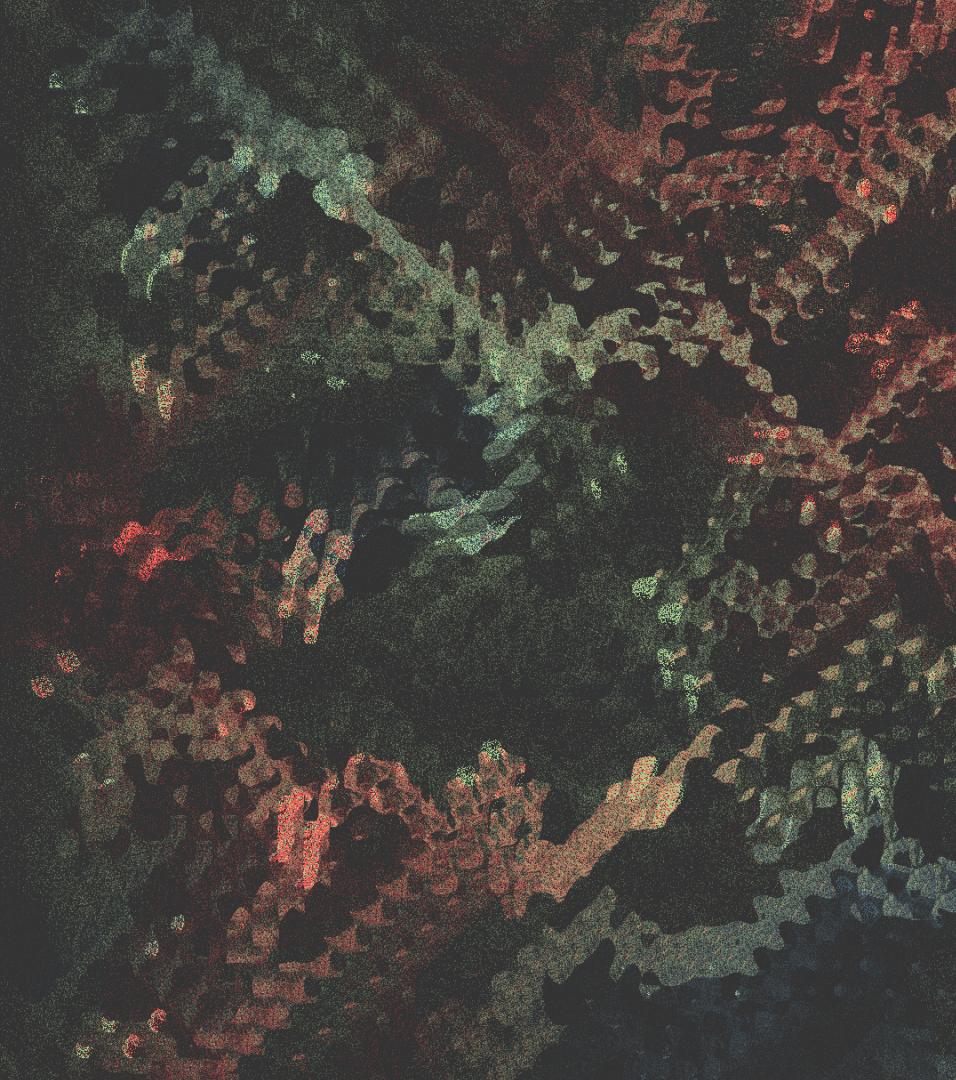




# Bevegelse på støyfelt perlin.js







# La oss komme i gang!

1. Gå til <https://github.com/plusk/generative-workshop/> og hent koden som en zippet mappe eller via Git
2. Følg instruksene for sette opp prosjektet med npm (eller uten)

Hvis du ønsker å printe ut et bilde kan du gjøre som følger:

1. Pass på at «PRINT\_MODE» er satt til «true» på toppen av JavaScript-filen for å få god oppløsning!
2. Klikk på bildet i nettleseren for å laste det ned
3. Send bildefilen til ???