

# Yaratıcı Kodlama Atölyesi

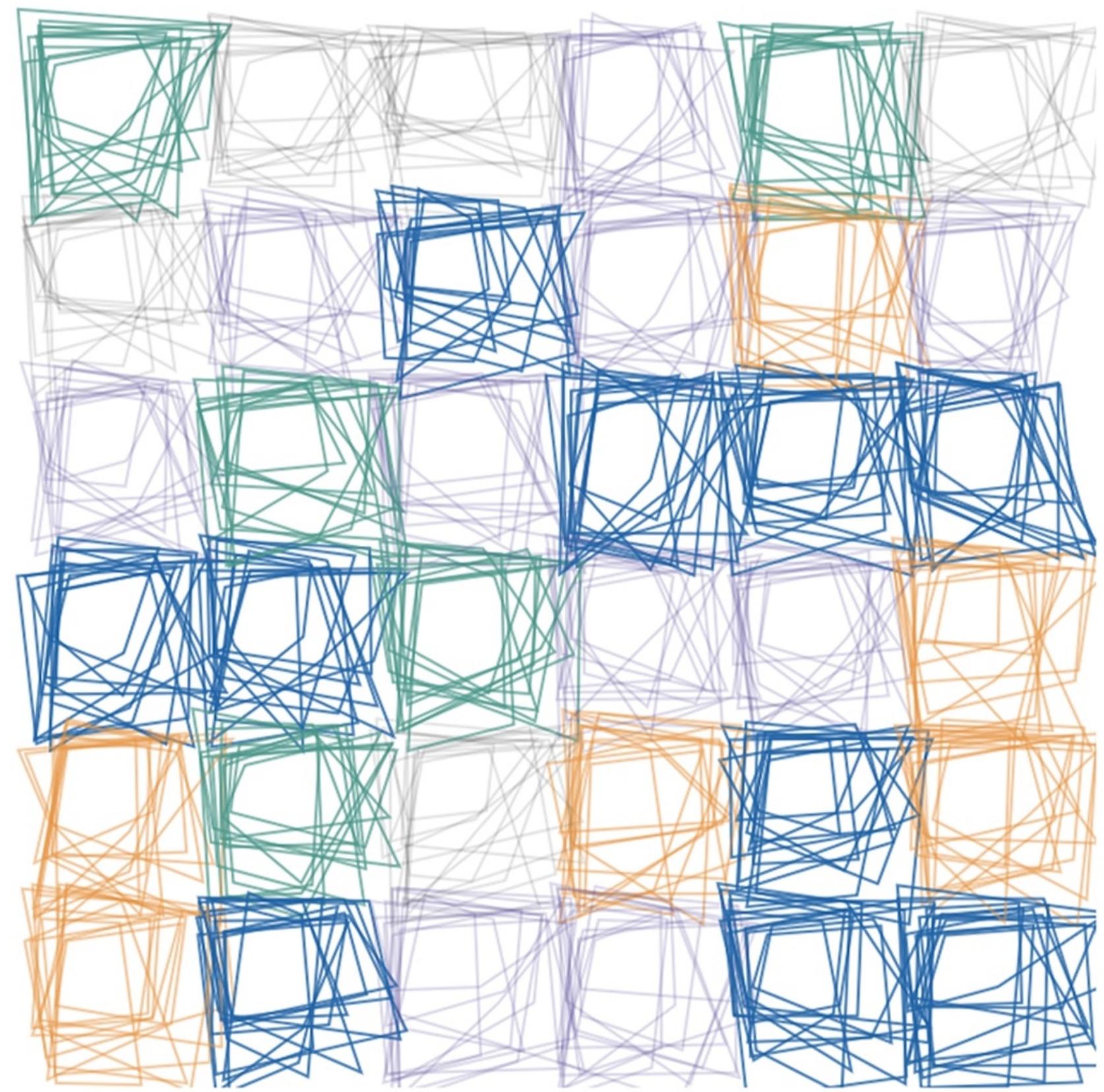
## Dijital Sanat

HOSGELDİNİZ!

Beklerken tanışalım mı?

Aşağıdaki linkteki (ya da QR'daki)  
soruları cevaplayabilir misiniz?

<https://padlet.com/bilg1/pera-da-yaratici-kodlama-at-lyesi-lhbe45t4bf3lit4c>



## Menüde neler var:

- 10 dk: Tanisma
- 10 dk: dijital sanata giriş
- 10 dk: p5'e baslangic
- 50 dk: cizim
- 10 dk: paylaşım ve sorular

[https://github.com/tbilgin/Yaratici\\_Kodlama/](https://github.com/tbilgin/Yaratici_Kodlama/)

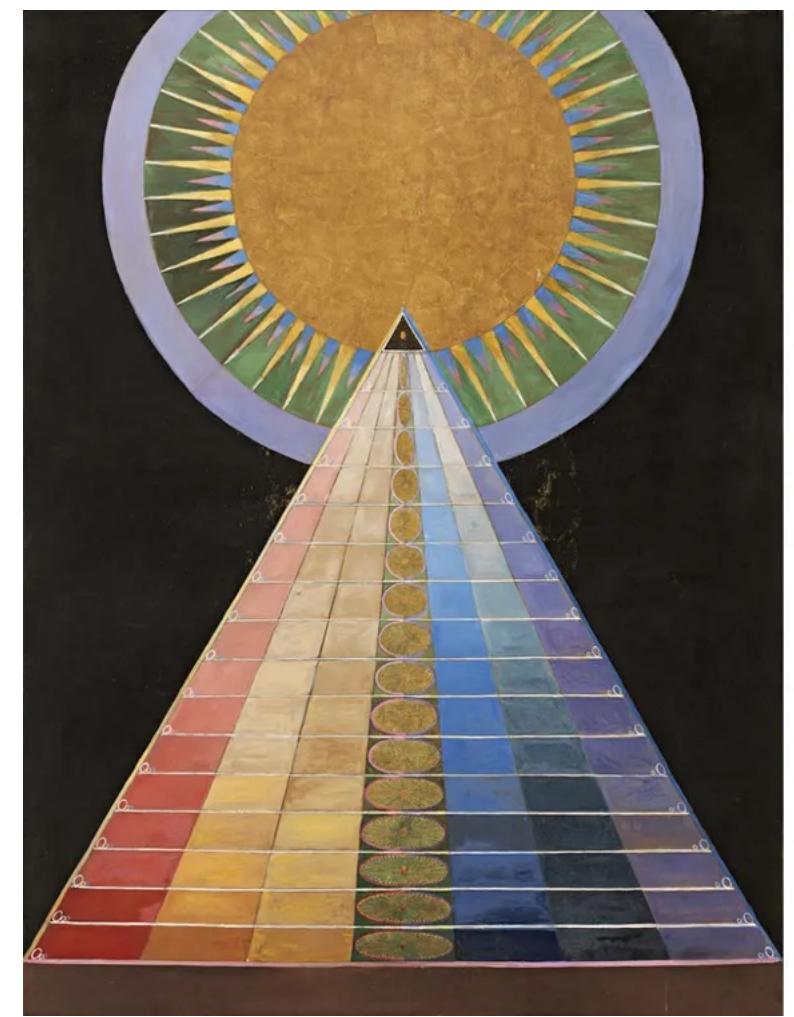
# Neden sanatta kodlama?

Otonomluktan uzaklaşma ihtiyacı

- ruhanilik
- soyut dışavurumculuktaki kendiliginlenlik
- canlılarla etkileşim



Antoine Bridier-Nahmias



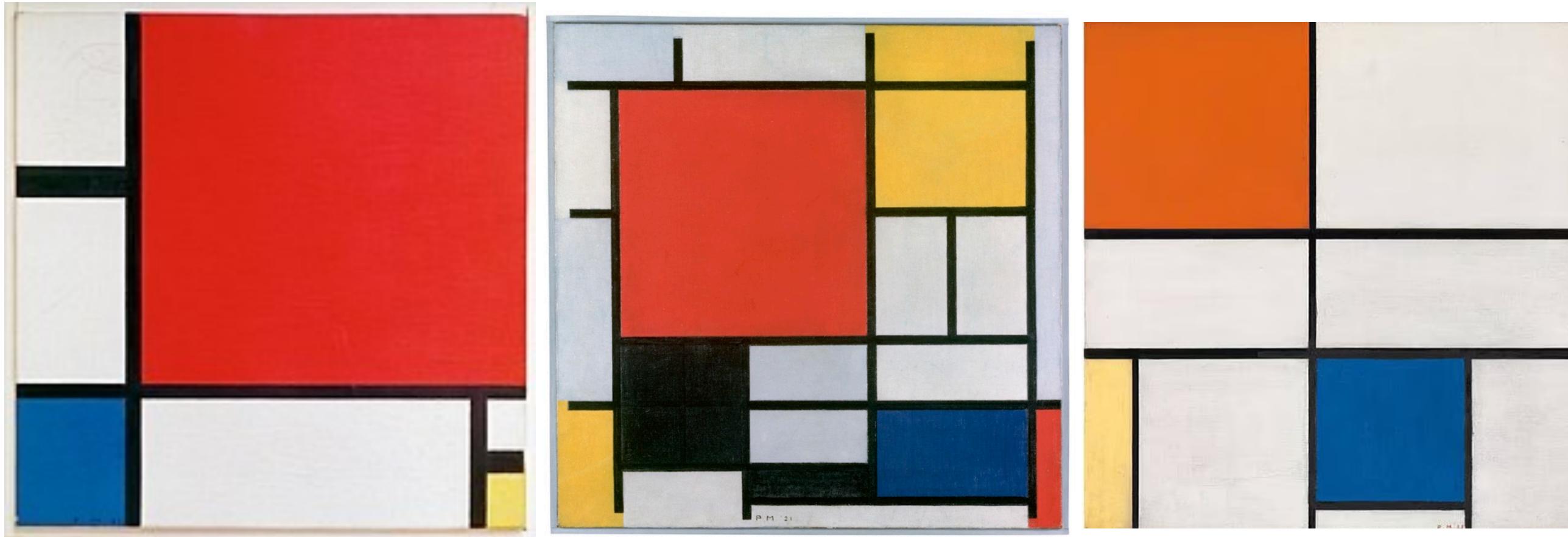
Hilma af Klint



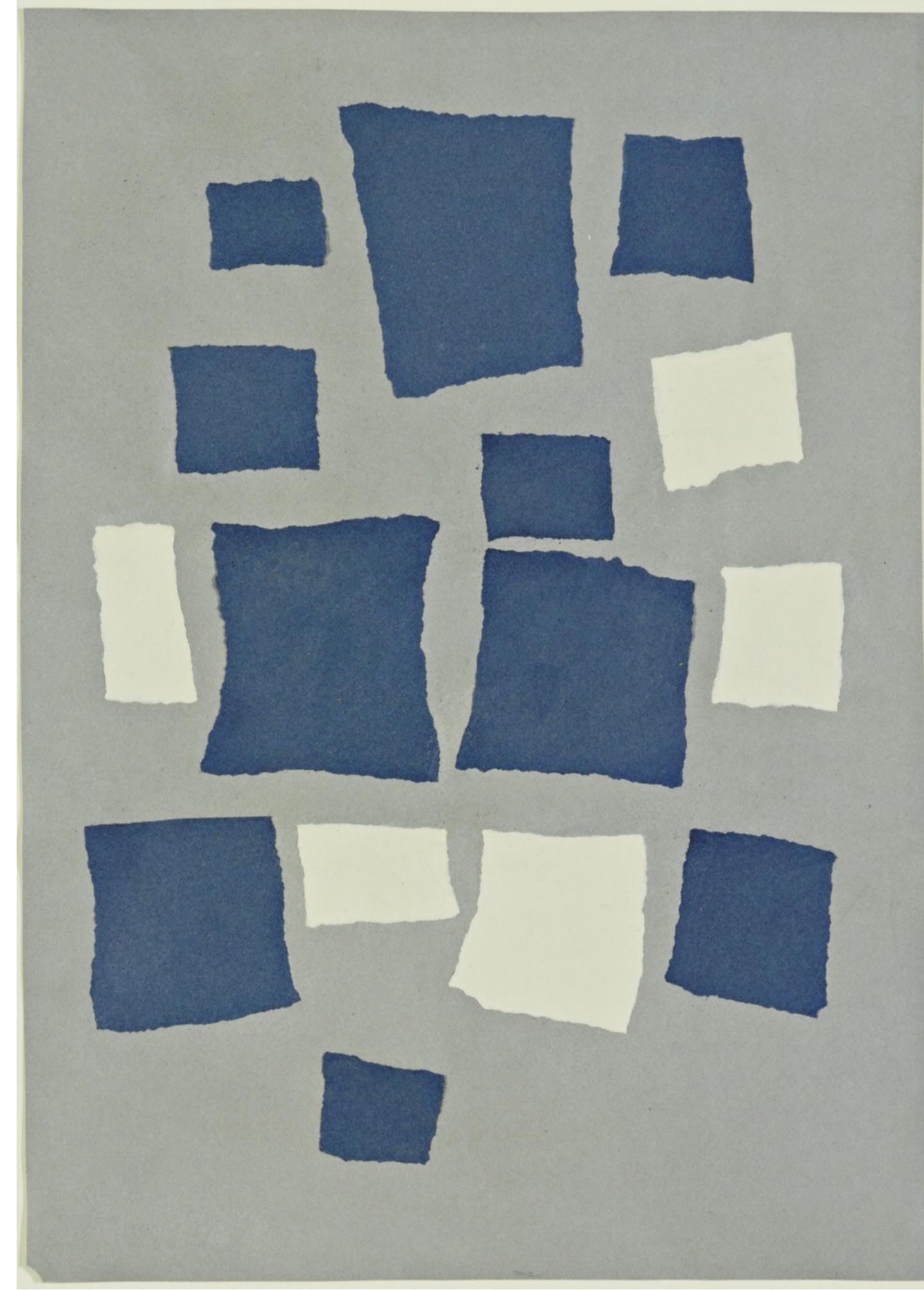
Jackson Pollock

# Karar mekanizması nasıl işliyor?

- o sans (dadaizm)
- o Kurallar



Mondrian



Hans Arp, 1916-17

# Sergi



## HESAPLAR ve TESADÜFLER

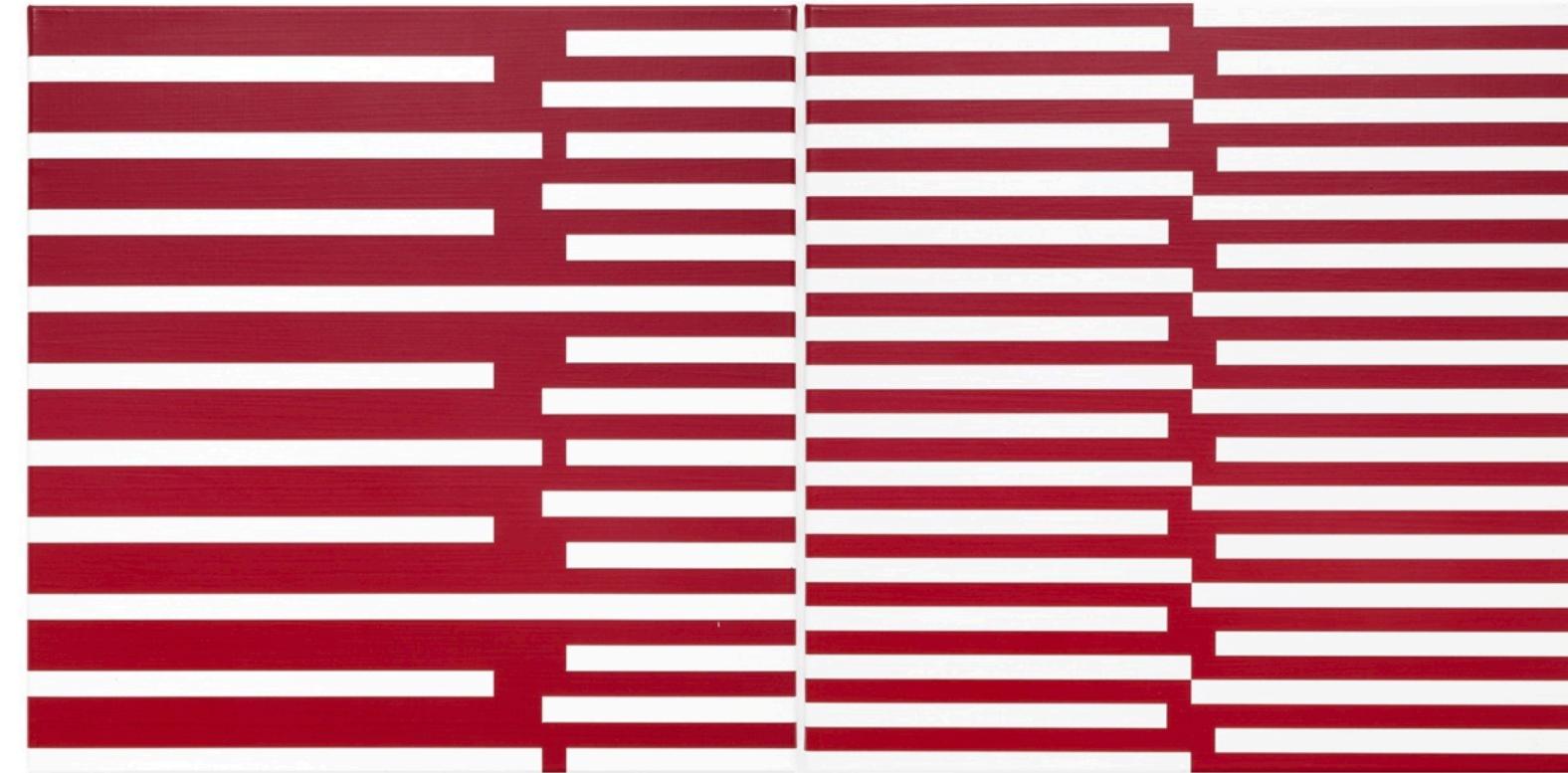
Dóra Maurer + Vera Molnár + Gizella Rákóczi

Macaristan Ulusal Bankası Koleksiyonu'ndan Algoritma Sanatı

## Vera Molnar (d. 1924, Budapeste)



- o bilgisayarda sanatin öncülerinden
- o algoritmik rastlantısallık
- o düzen ve düzensizlik, yapı ve özgürlük gibi ikilikler



tahta üzerine akrilik,  
2010-2012, 2 parça, 50 x 50 cm

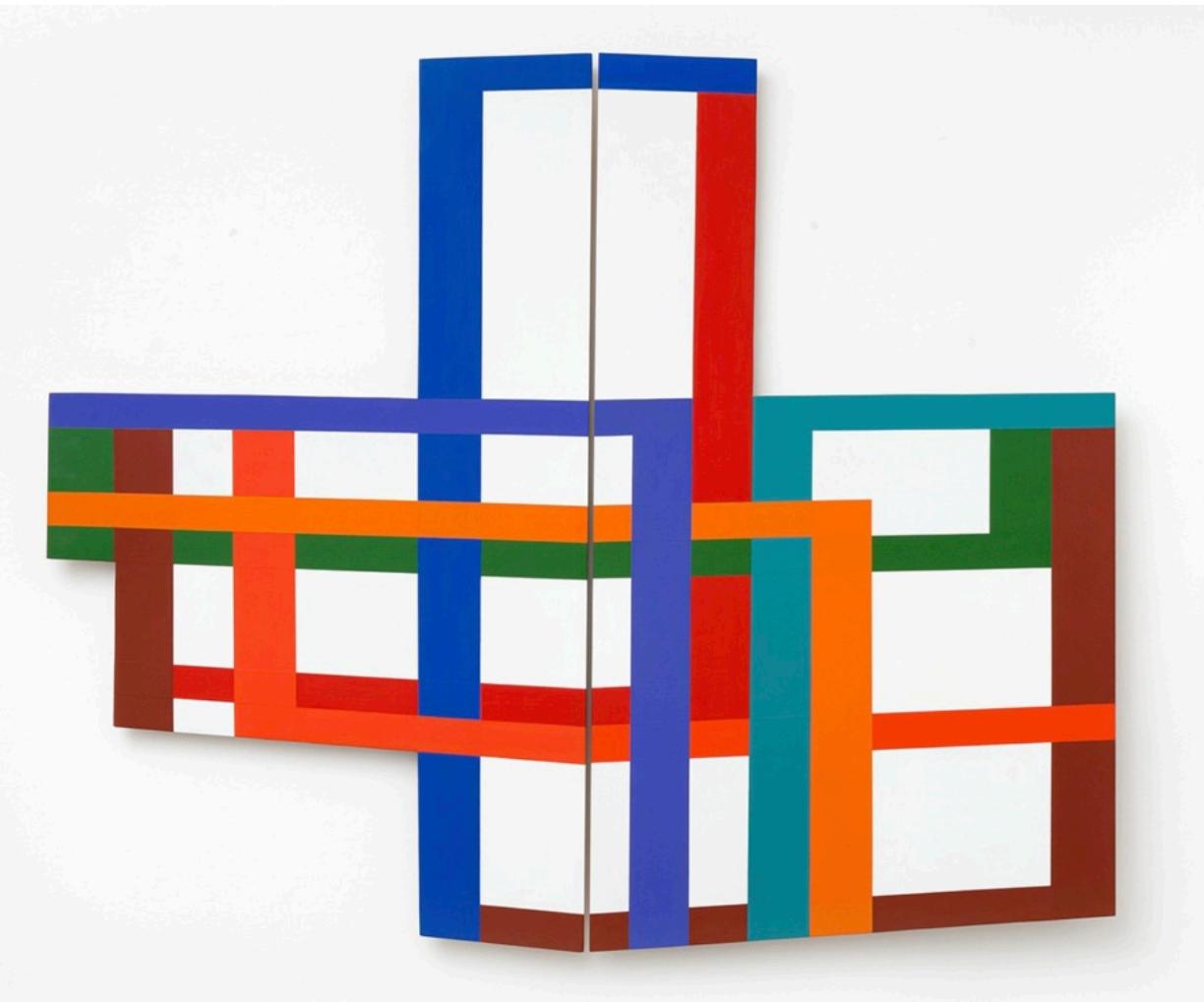


yağlı boya, 162-1964,  
110 x 110 cm

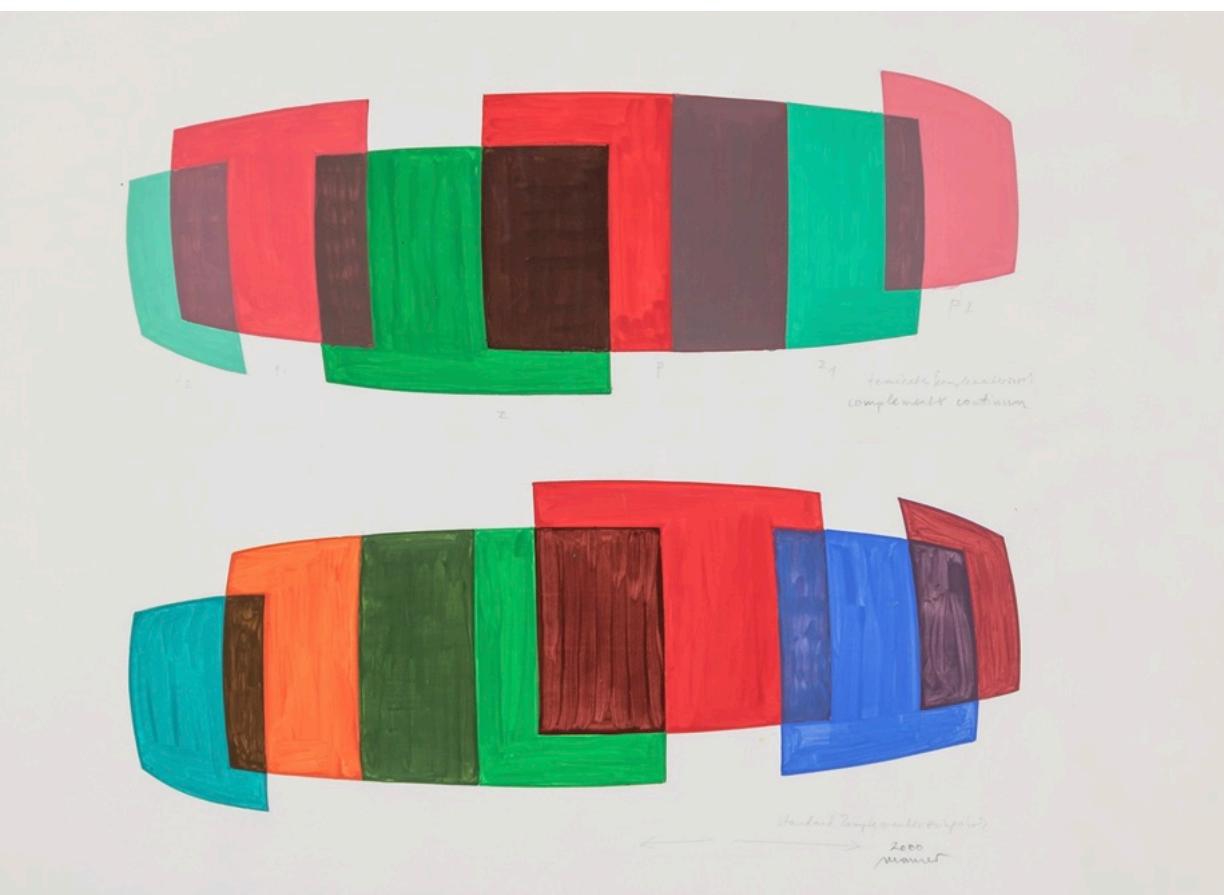
## Dora Maurer (d. 1937, Budapeste)



- o matematik, mantik, geometri
- o sistematikleştirme, düzenleme, kümleme



tahta üzerine akrilik, 1990,  
114 x 133 cm

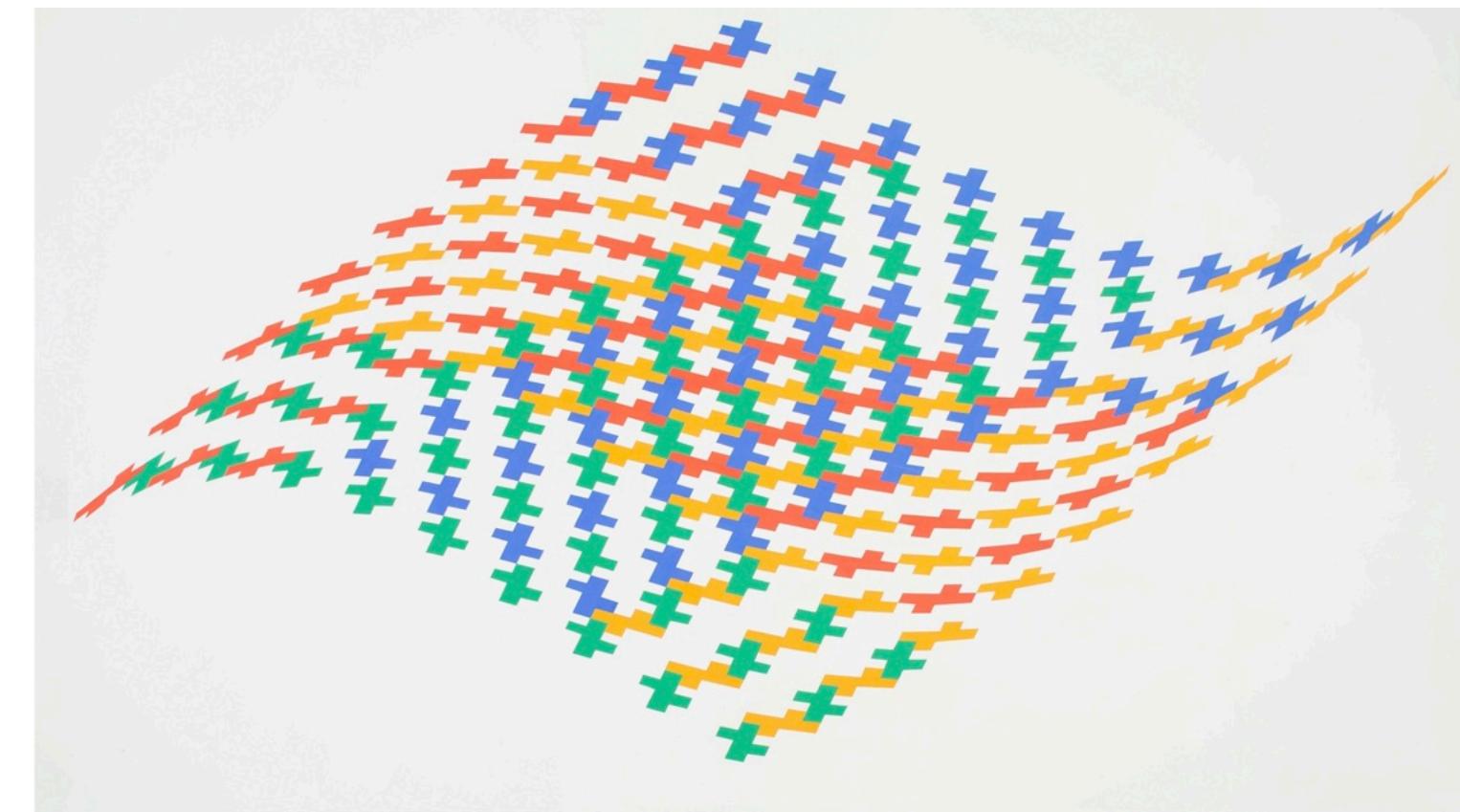


kağıt üzerine tempera, 2000,  
48 x 61 cm

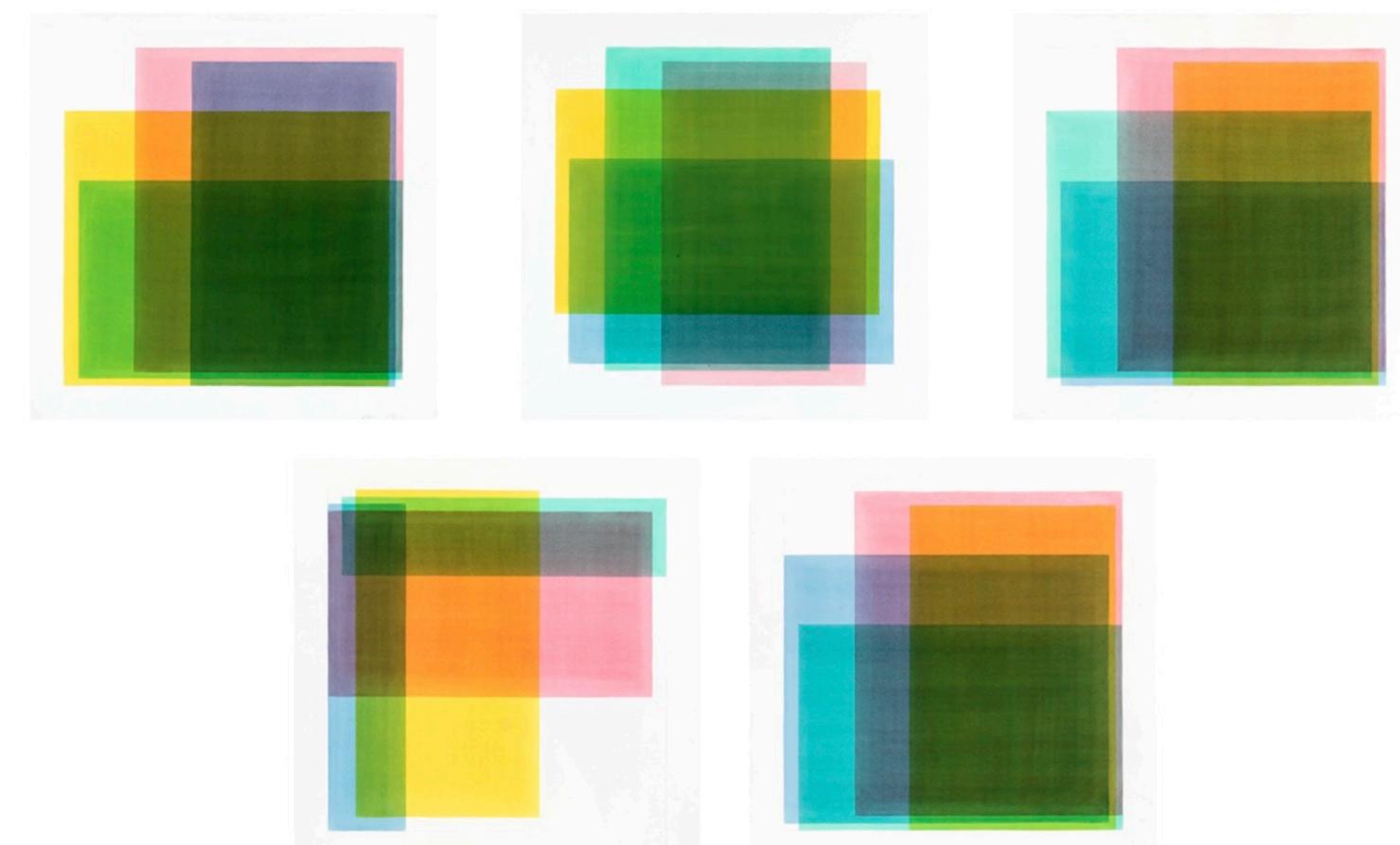
## Gizella Rakoczy (d. 1947, Budapeste)



- olasılık, geometri: spirallerin hareketleri ve seri olasılıklarının modelleri
- matematik: Fibonacci serisinin formülünü kullanarak şeffaf boyanın tonlarını katmanlaştırmaya başladığı sulu boya resimler



kağıt üzerine tempera, 1987,  
117 x 198.5 cm



sulu boya, 2002, 5 parça,  
69 x 69 cm

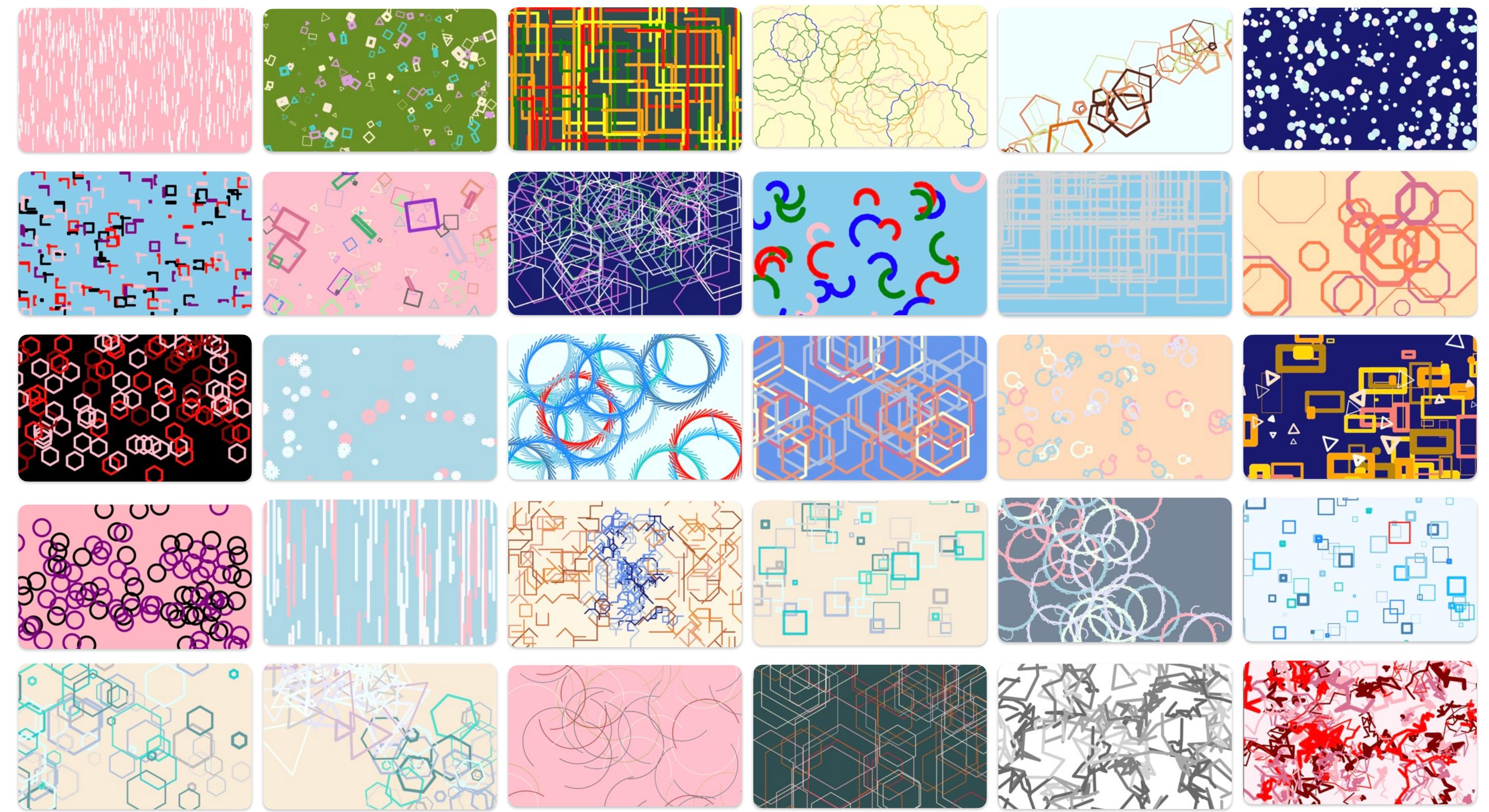
# Yapay Zeka ile Sanat



Refik Anadol,  
Unsupervised, MoMA, 2023

# Yaratıcı kodlama

- o p5 <https://p5js.org/>
- o python vs.



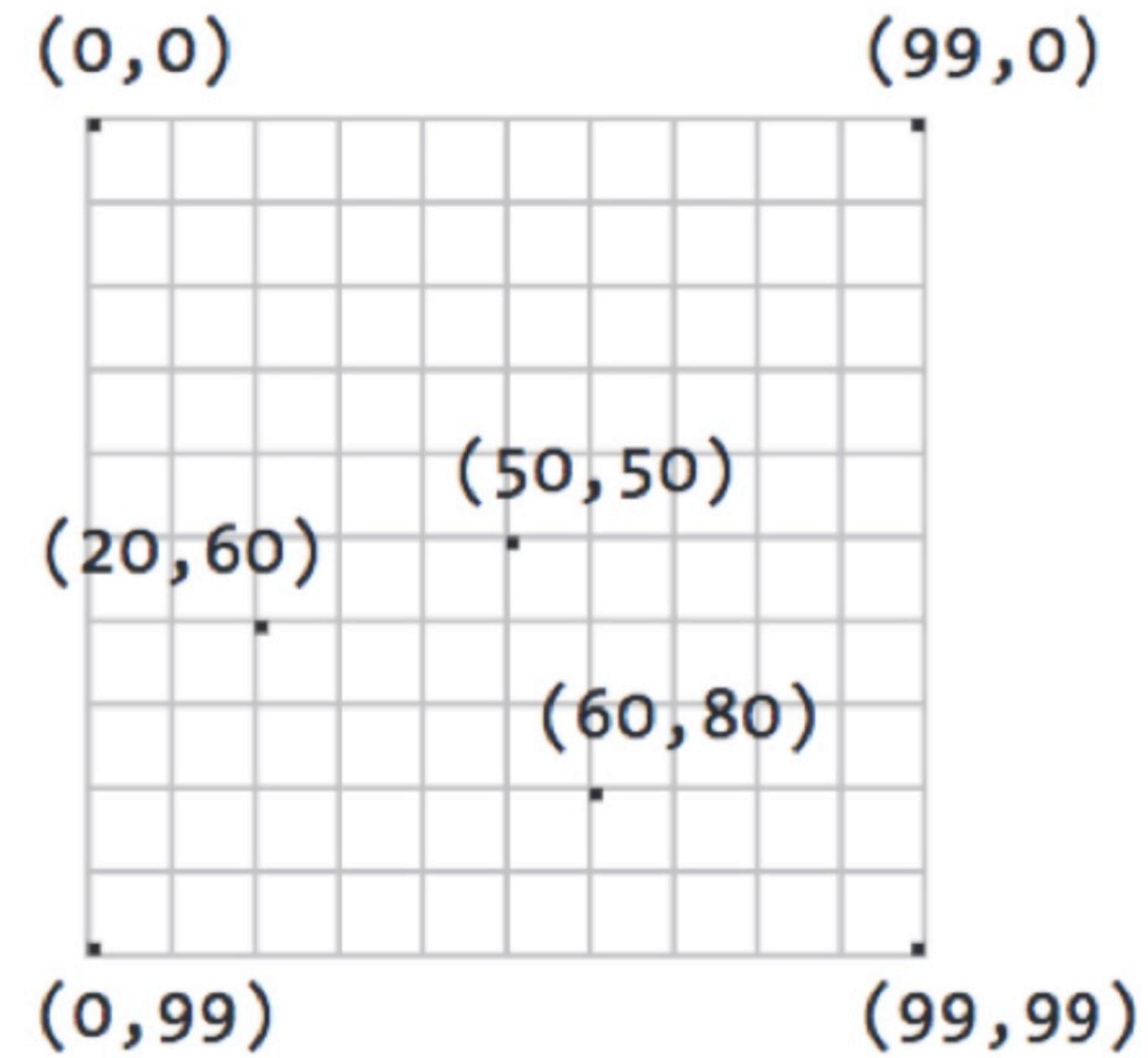
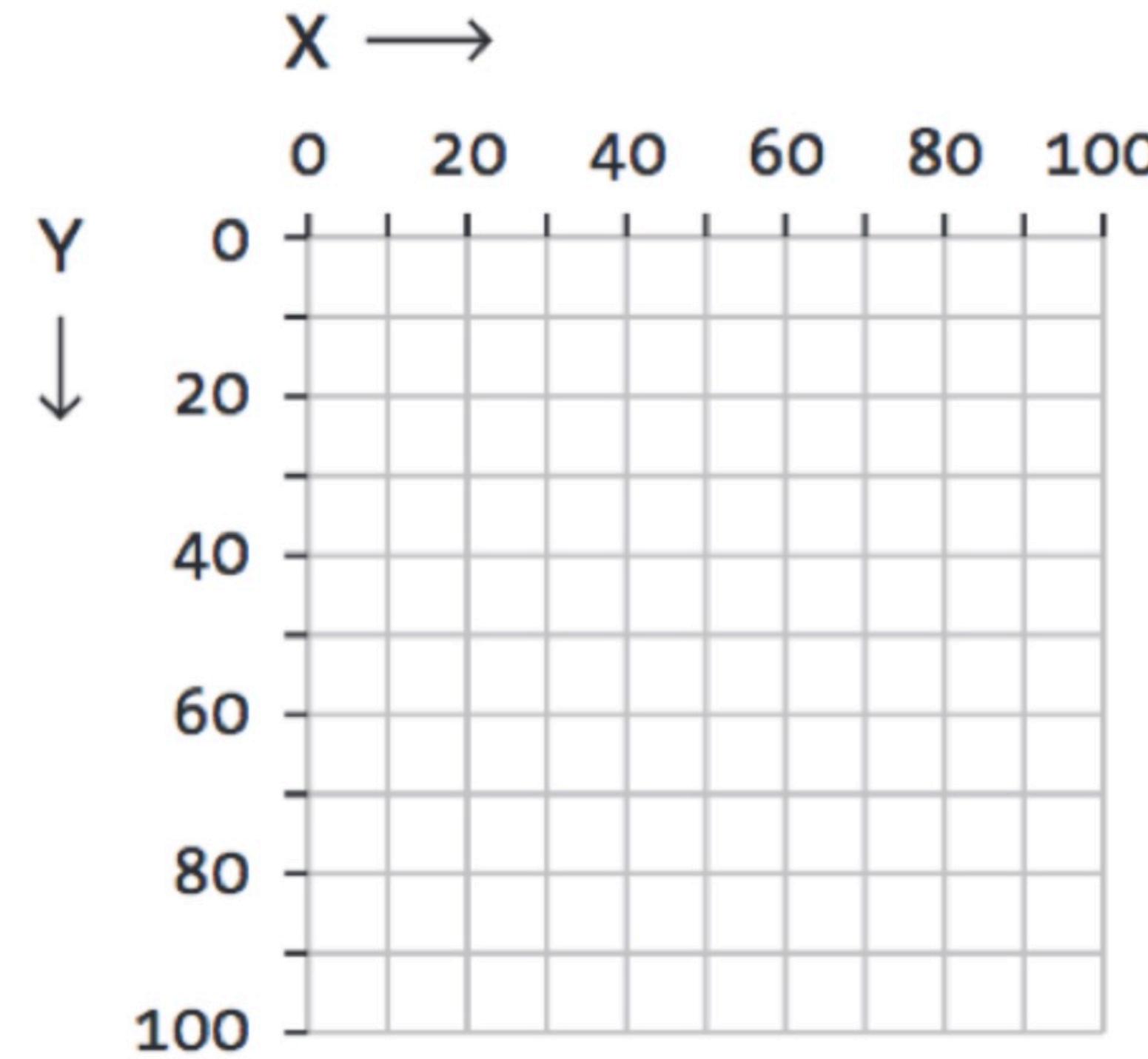
# p5

The screenshot shows the p5.js sketch editor interface. The title bar says "p5\*". The menu bar includes "File", "Edit", "Sketch", and "Help". The top right shows "English" and "Hello, tbilgin!". Below the menu is a toolbar with a play button, a square, and other icons. It also has an "Auto-refresh" checkbox and the text "Pera Atölye by tbilgin". A gear icon is in the top right corner. The main area has tabs for "sketch preview" and "sketch.js". The code editor shows "sketch.js" with the following code:

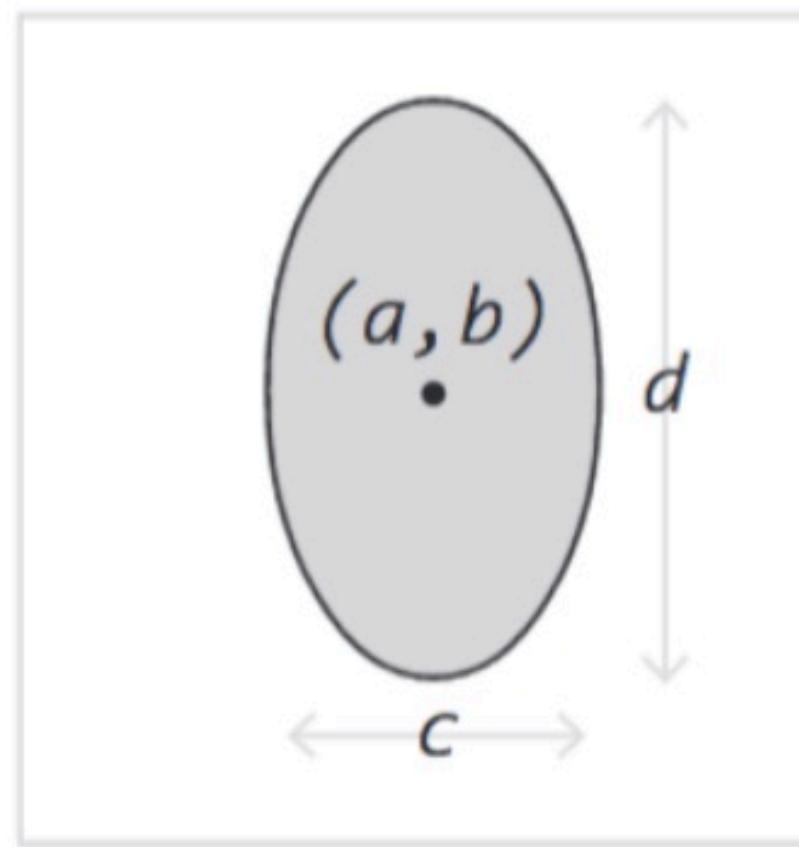
```
1 function setup() {
2   createCanvas(400,400);
3 }
4
5 function draw() {
6   background(255);
7
8   rect(200,100,200,50)
9 }
10 }
```

To the right is a preview window showing a white canvas with a single black rectangle centered at (200, 100) with dimensions 200x50. At the bottom left is a "Console" tab and a "Clear" dropdown. The bottom right of the editor has a small "Pera Atölye" watermark.

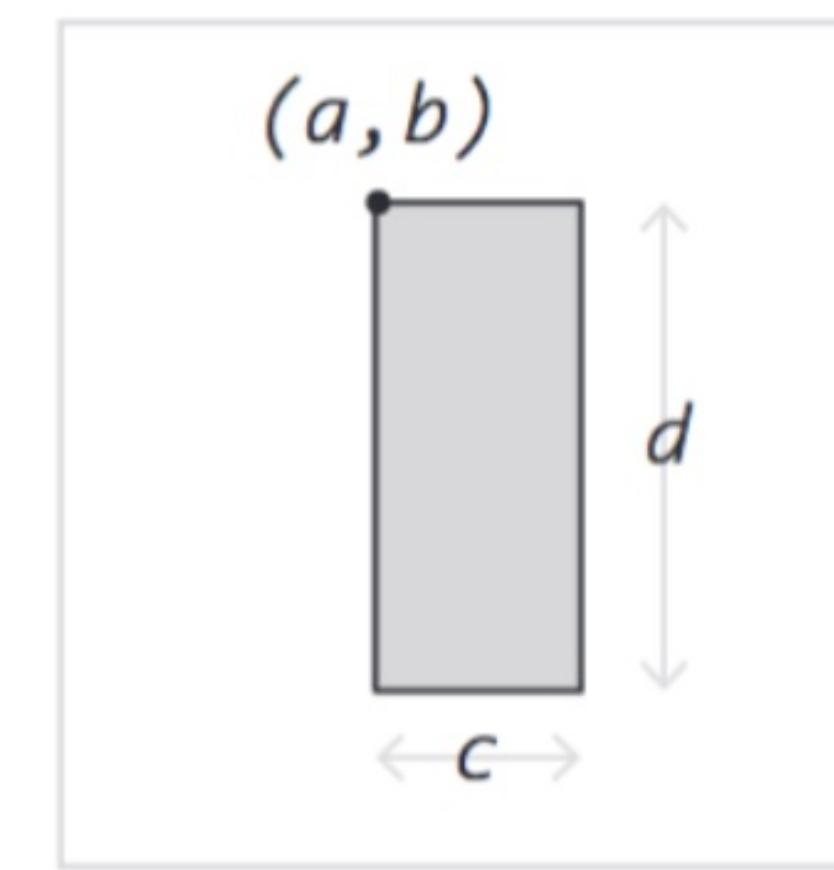
# Dijital Kanvas



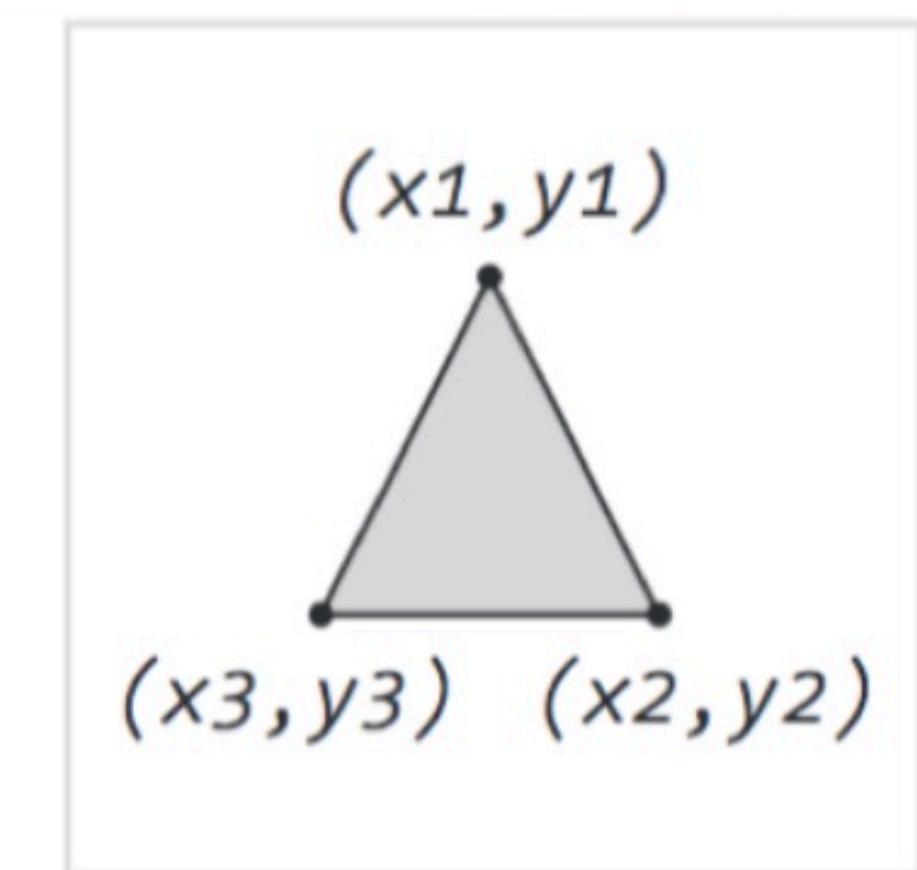
# Dijitalde Sekiller



*ellipse(a, b, c, d)*



*rect(a, b, c, d)*



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

# Dijitalde Sekiller

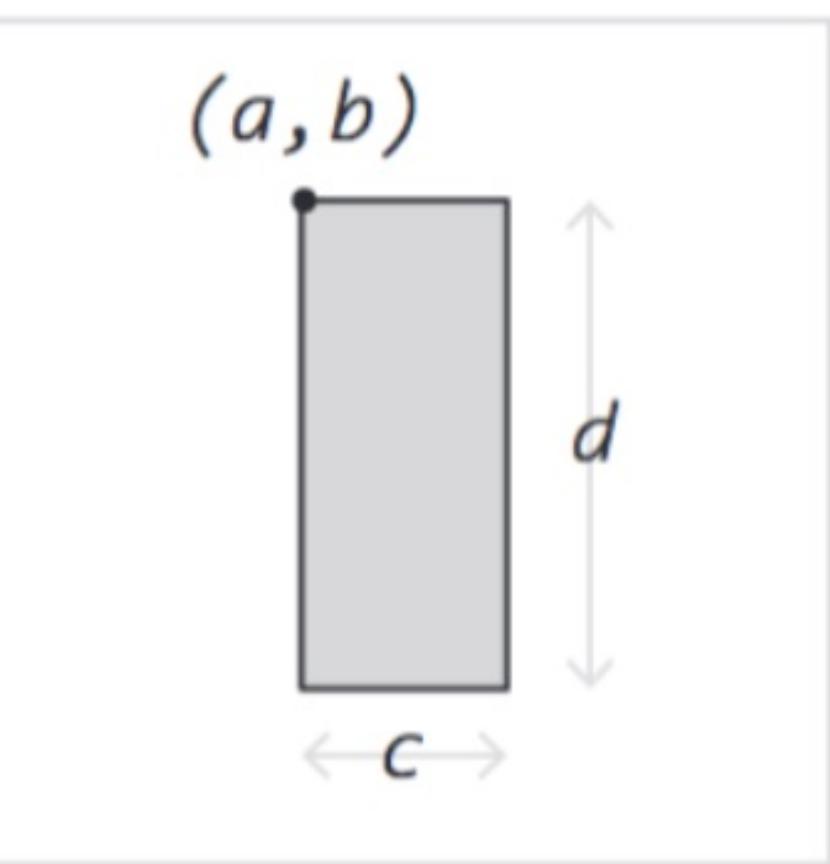
p5\* File ▾ Edit ▾ Sketch ▾ Help ▾ English ▾ Hello, tbilgin! ▾

Auto-refresh Pera Atölye by tbilgin

sketch.js Saved: 25 seconds ago Preview

```
1 function setup() {  
2   createCanvas(400, 400);  
3 }  
4  
5 function draw() {  
6   background(255);  
7  
8   rect(0, 0, 200, 50)  
9 }
```

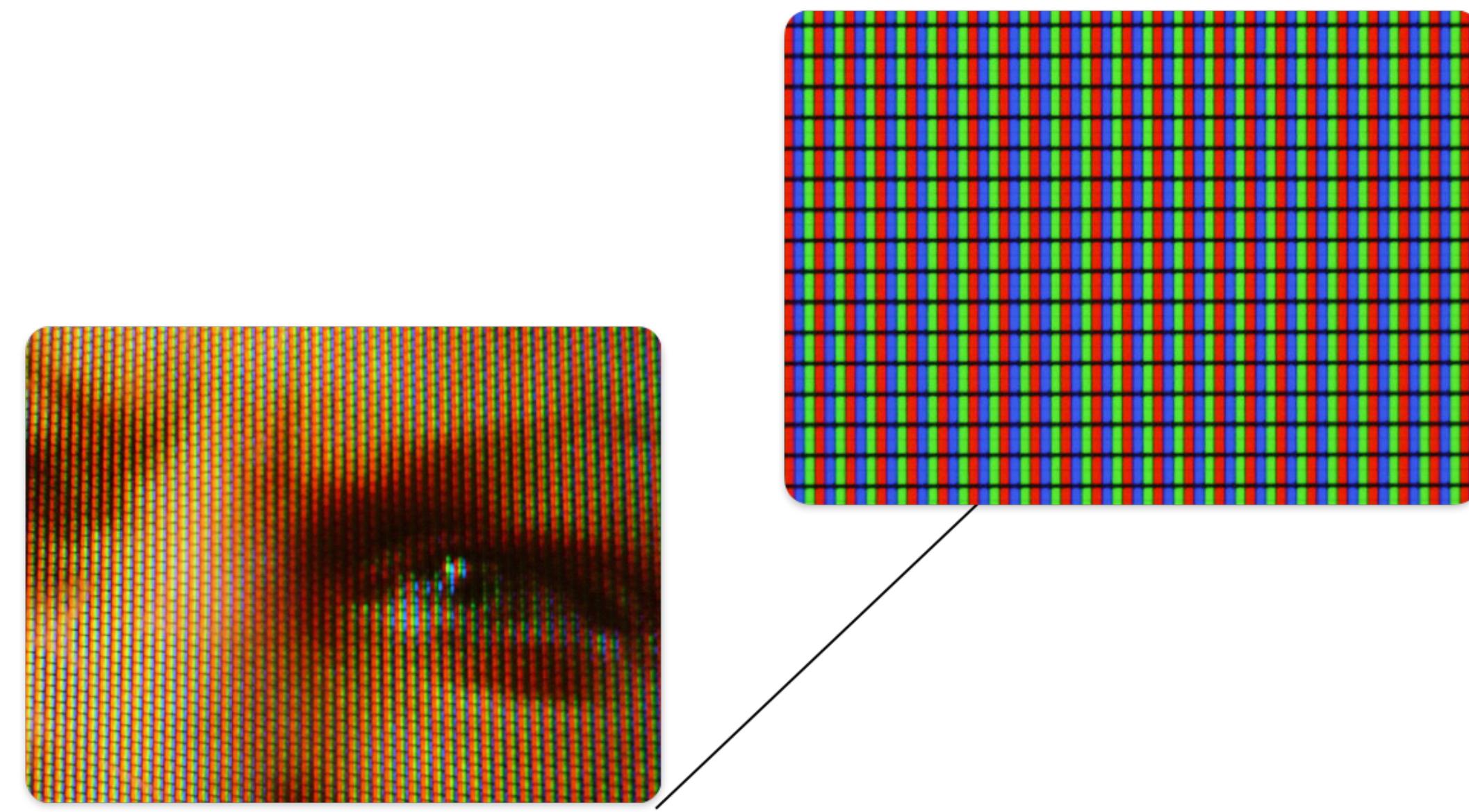
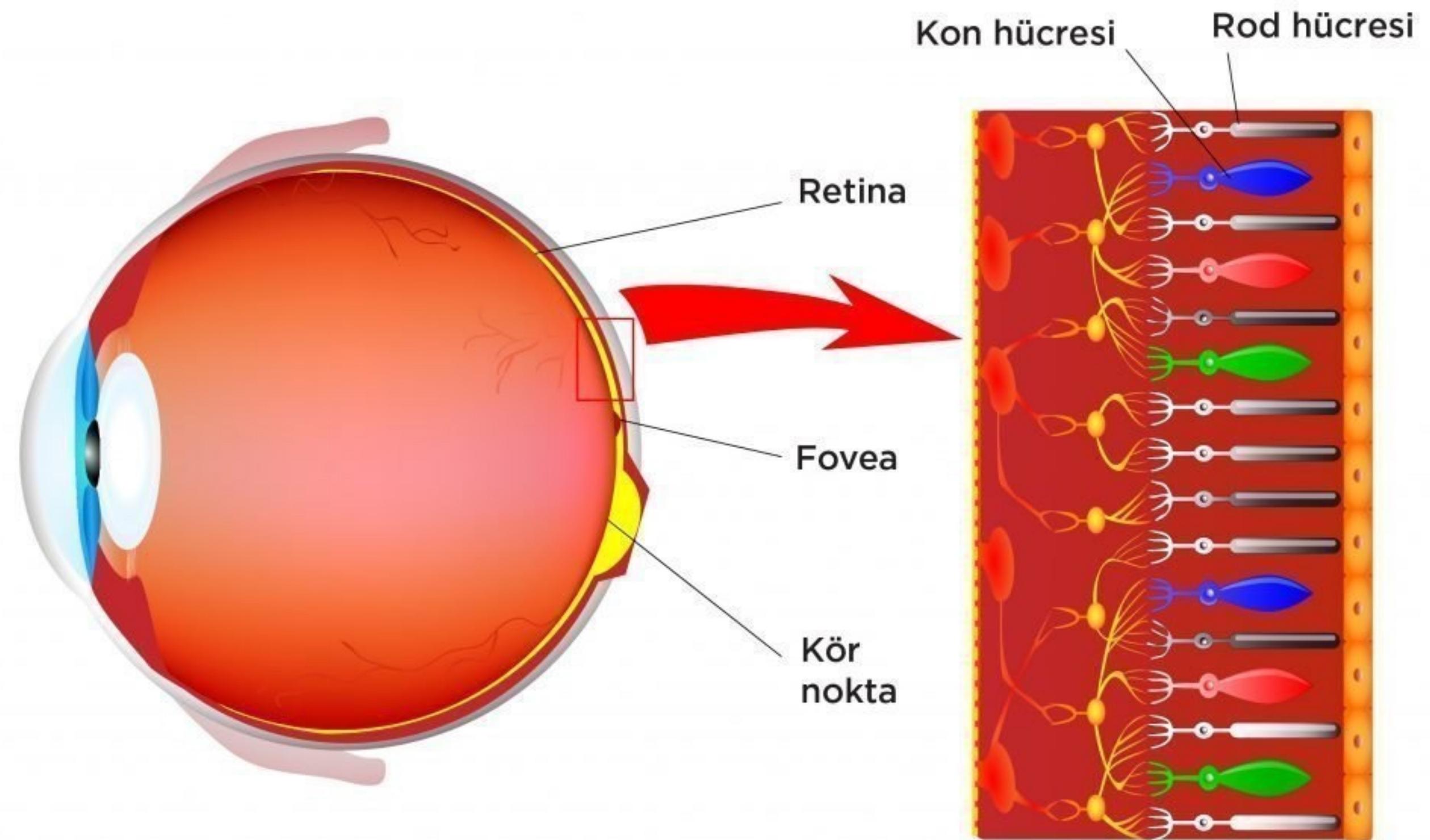
Console Clear ▾



$\text{rect}(a, b, c, d)$

# Dijitalde Renkler

## Nasıl görürüz?



# Dijitalde Renkler

p5\*

File ▾ Edit ▾ Sketch ▾ Help ▾

English ▾ Hello, tbilgin! ▾

Auto-refresh Pera Atölye by tbilgin

sketch.js

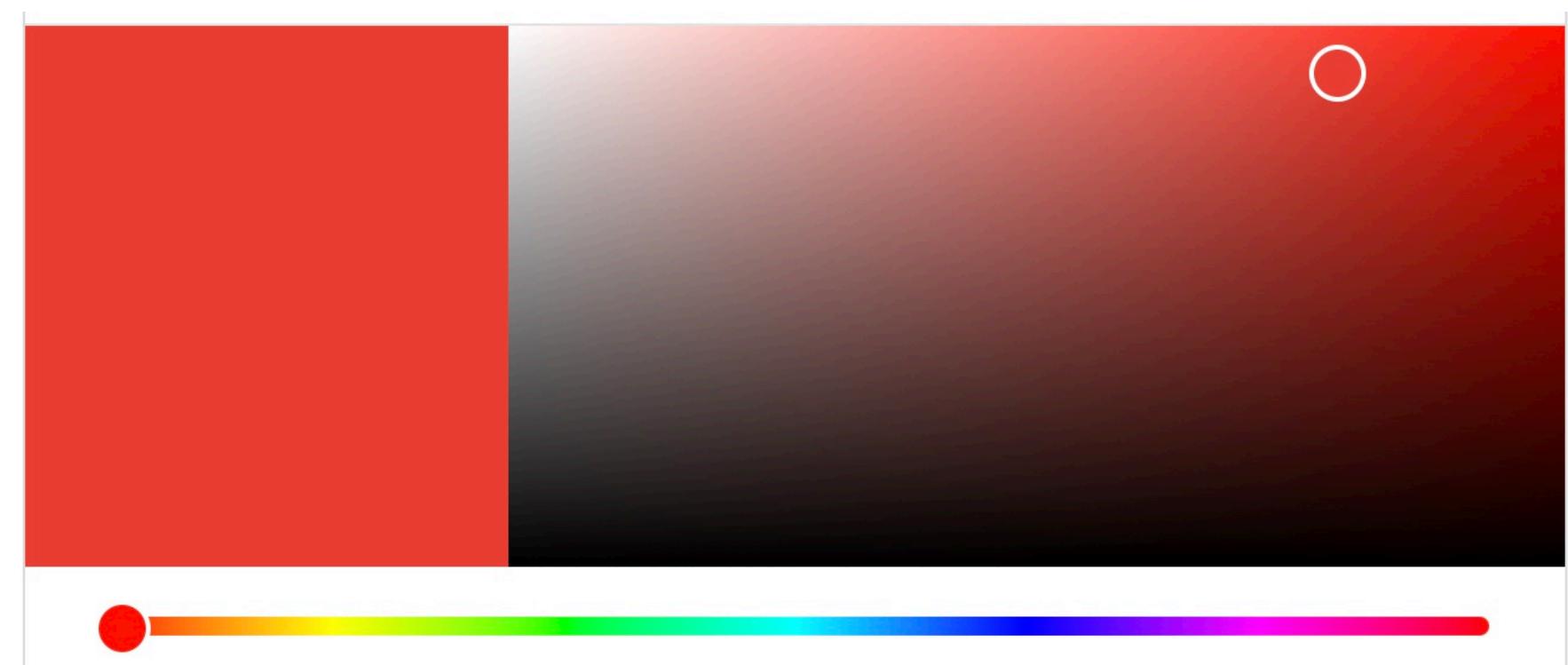
Saved: just now Preview

```
1 function setup() {  
2   createCanvas(400,400);  
3 }  
4  
5 function draw() {  
6   background(235,64,52);  
7  
8   rect(0,0,200,50)  
9 }  
10 }
```

Console Clear ▾



renk seçici



RGB  
235, 64, 52

# Haydi biraz oynayalım!

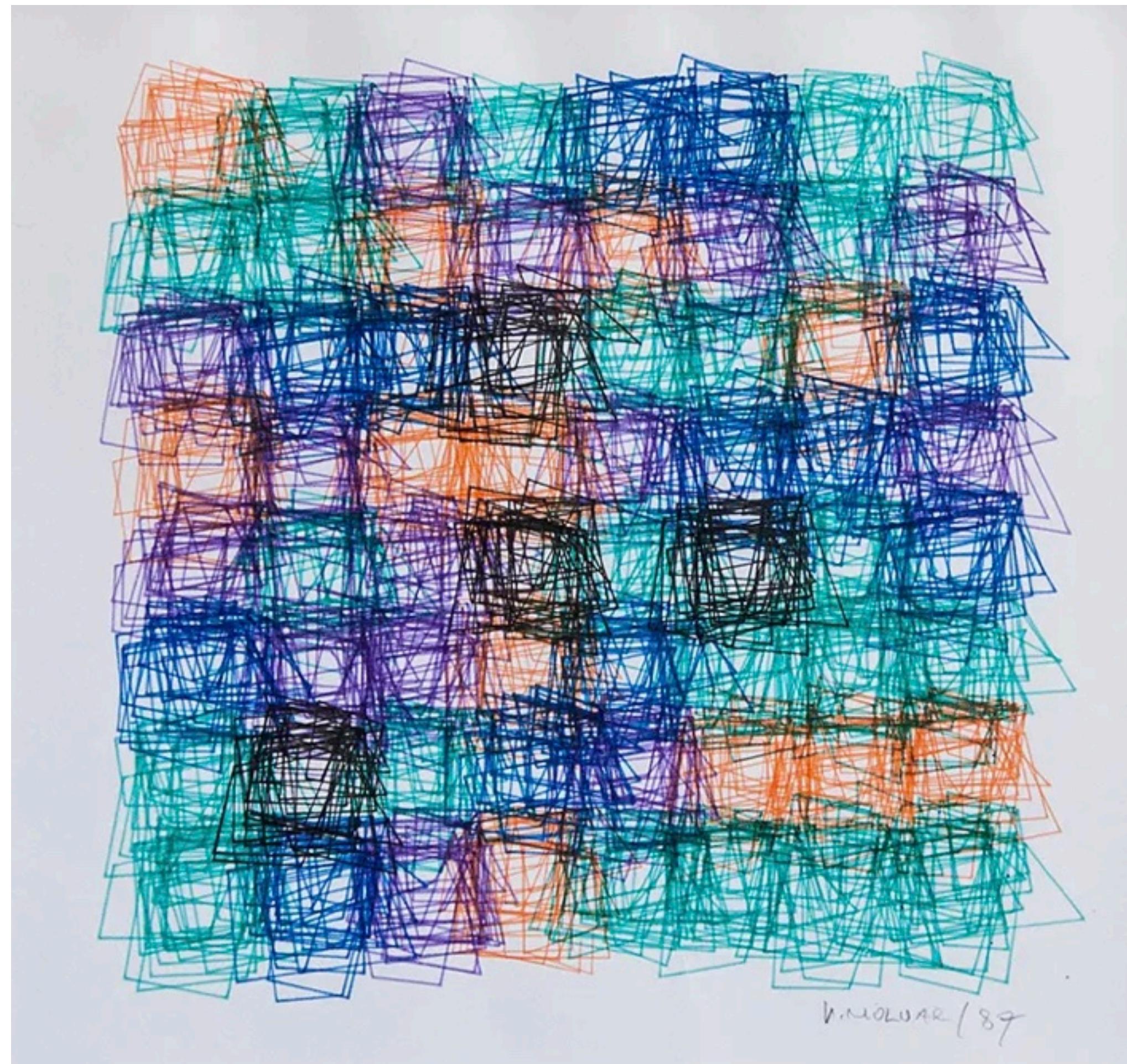
<https://editor.p5js.org/tbilgin/sketches/xdqGbMlXG>

The screenshot shows a p5.js code editor interface. At the top, there's a header with a back arrow, the file name 'sketch.js', and a save status 'Saved: 35 seconds ago'. To the right is a preview window labeled 'Preview'. The code area contains the following script:

```
1 function setup() {
2   createCanvas(400,400);
3 }
4
5 function draw() {
6   background(235,64,52);
7
8   fill(10,10,200);
9   rect(0,0,200,50);
10
11  fill(20,80,120);
12  rect(100,30,20,150);
13
14  fill(120,180,20);
15  ellipse(150,100,50,50);
16
17
18 }
```

The preview window displays a red background with a blue rectangle at the top, a dark blue vertical bar on the left, and a green circle in the center.

# Atölye



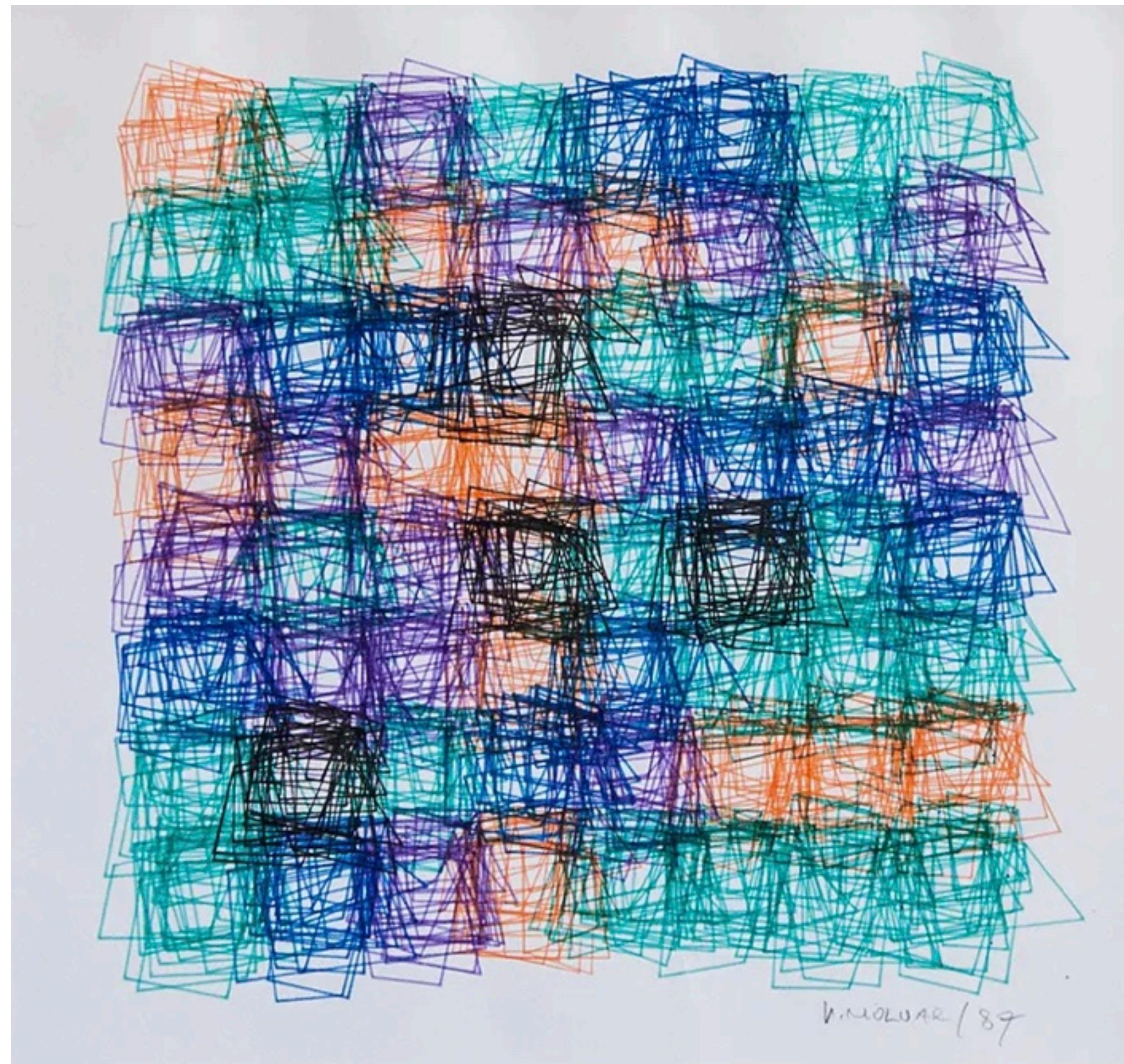
Bunu nasıl çizeriz?

Vera Molnar

Structure de Quadrilatères (Kare Yapılar)

kağıt üzerine tükenmez kalem, 1987, 29.5 × 30.9 cm

# Atölye



Bunu nasıl çizeriz?

- sekil
- renk
- tekrar
- rastgelelik
- kontrol
- kurallar

# Sekil

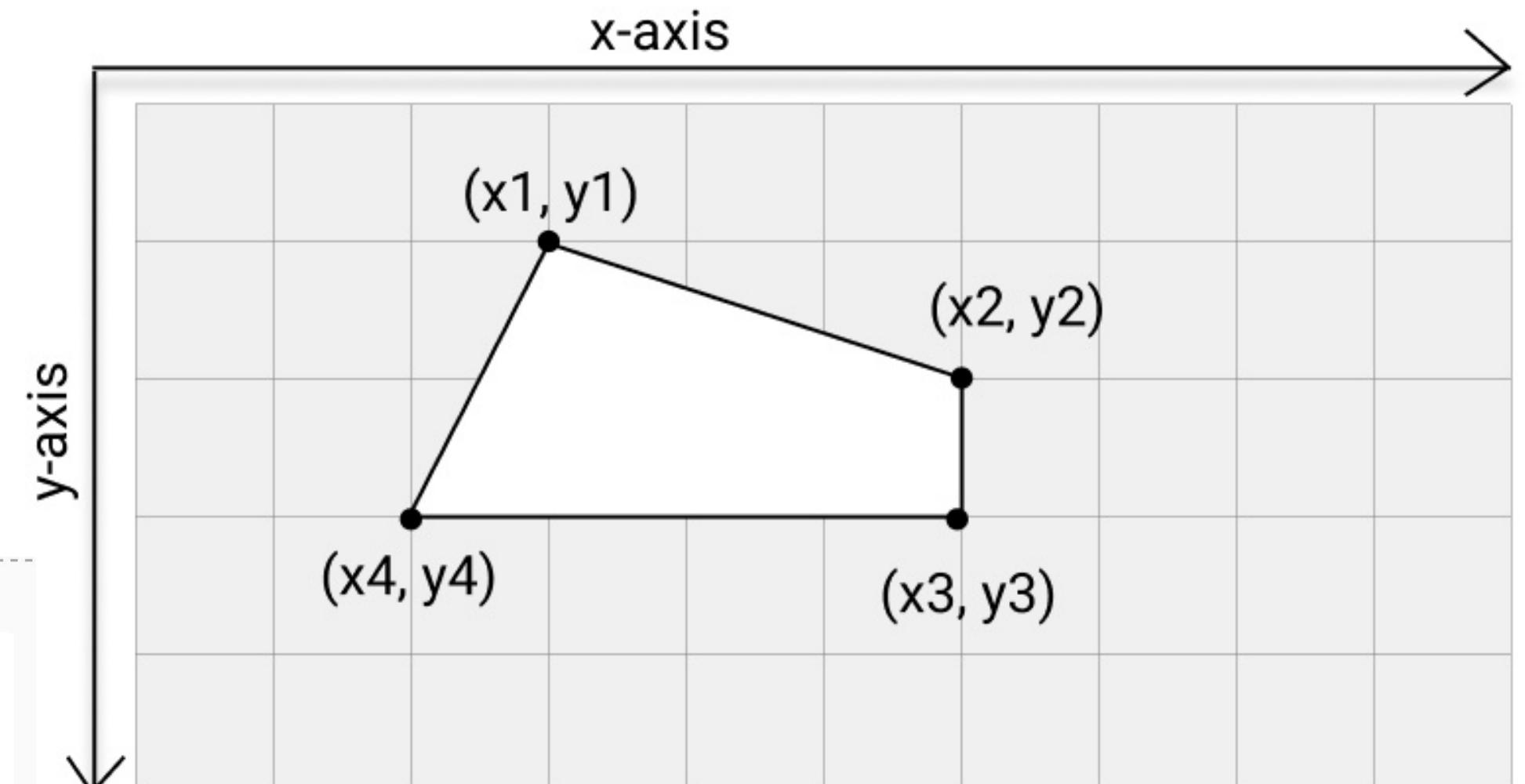
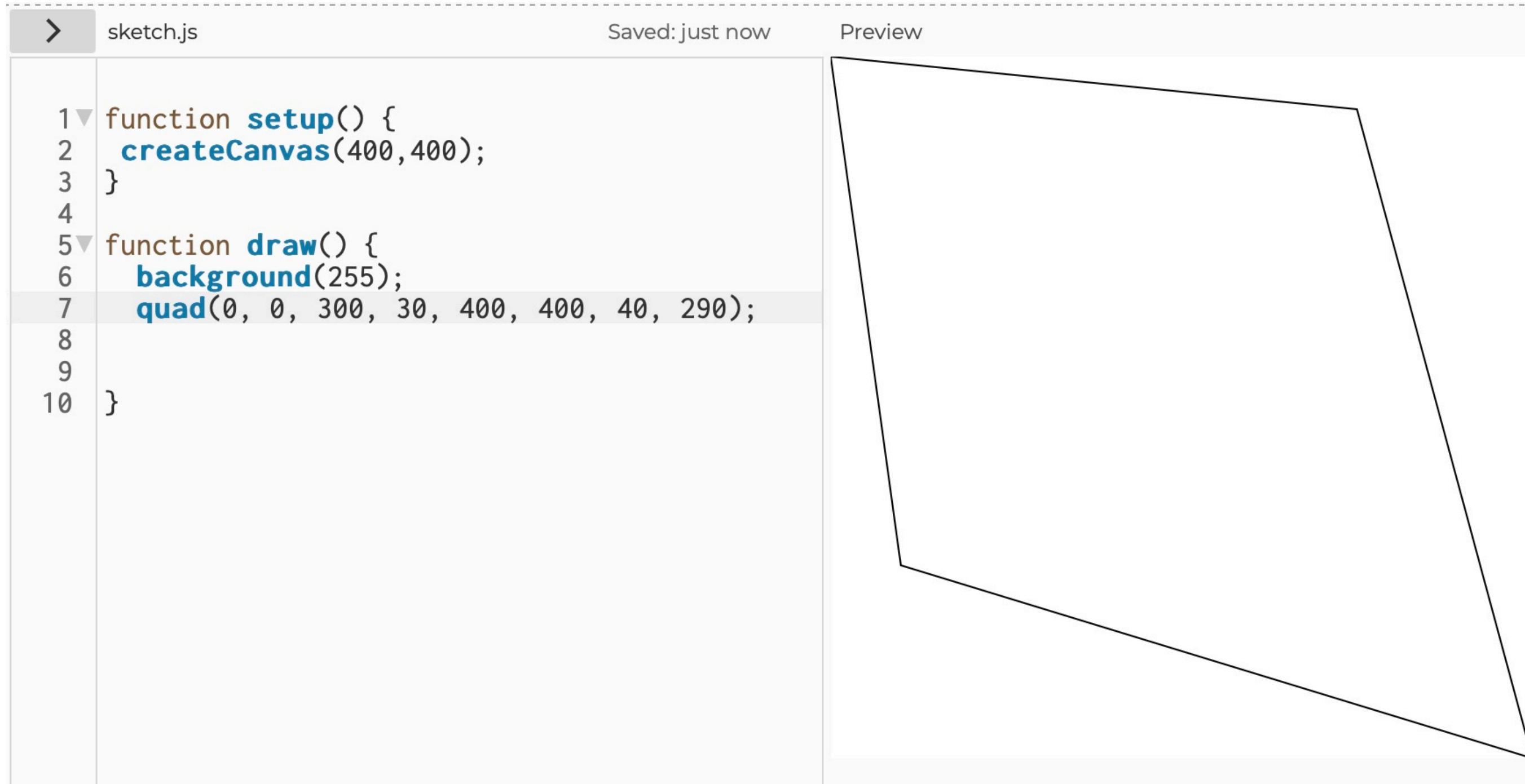
## Dörtgen: quadrilateral quad()

sketch.js

Saved: just now

Preview

```
1 function setup() {  
2   createCanvas(400,400);  
3 }  
4  
5 function draw() {  
6   background(255);  
7   quad(0, 0, 300, 30, 400, 400, 40, 290);  
8 }  
9  
10 }
```



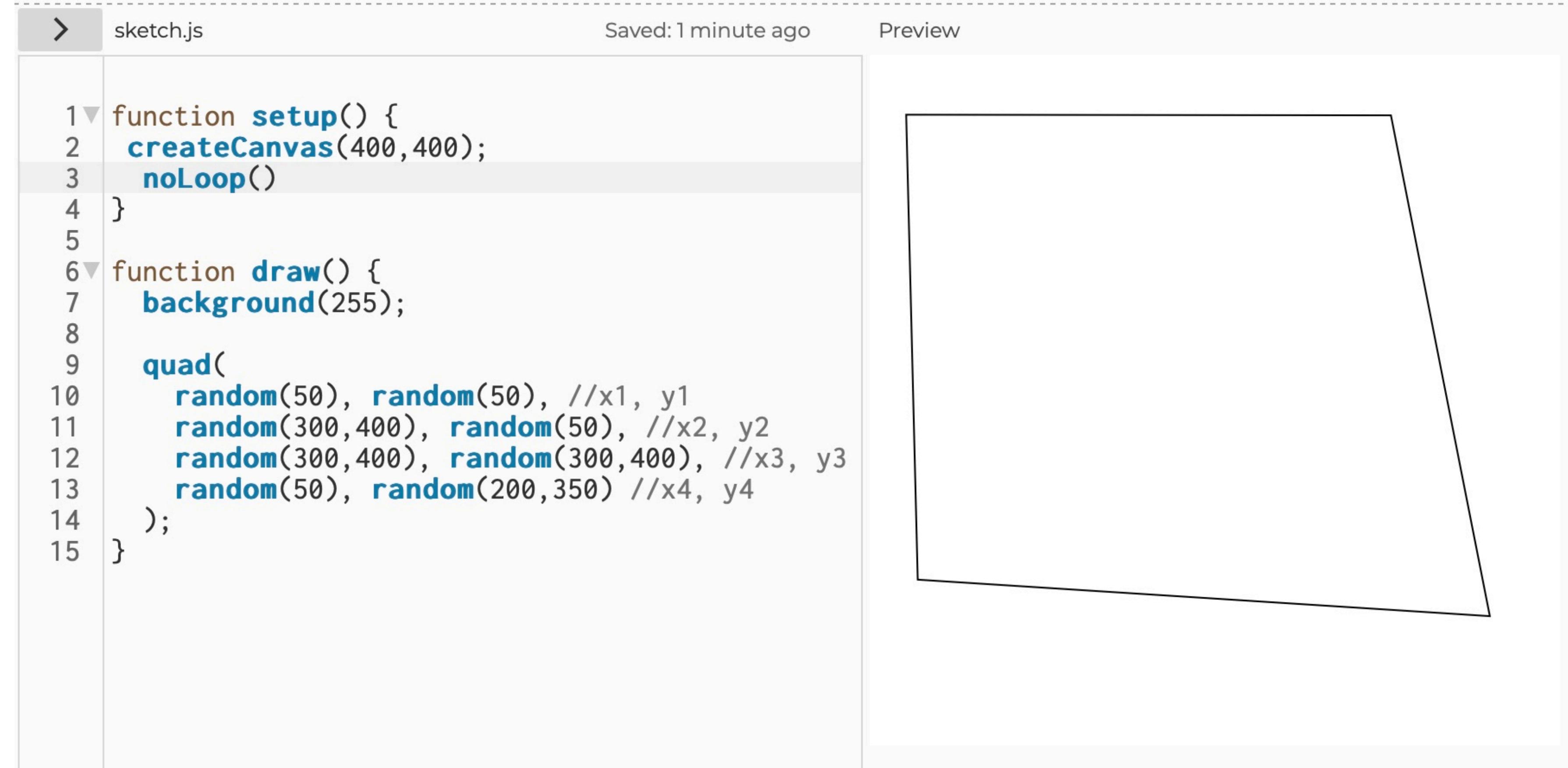
```
quad(x1, y1, x2, y2, x3, y3, x4, y4);
```

# Rastgelelik

<https://editor.p5js.org/tbilgin/sketches/2rMSk9mL7>

> sketch.js      Saved: 1 minute ago      Preview

```
1▼ function setup() {
2  createCanvas(400, 400);
3  noLoop()
4}
5
6▼ function draw() {
7  background(255);
8
9  quad(
10    random(50), random(50), //x1, y1
11    random(300,400), random(50), //x2, y2
12    random(300,400), random(300,400), //x3, y3
13    random(50), random(200,350) //x4, y4
14  );
15}
```

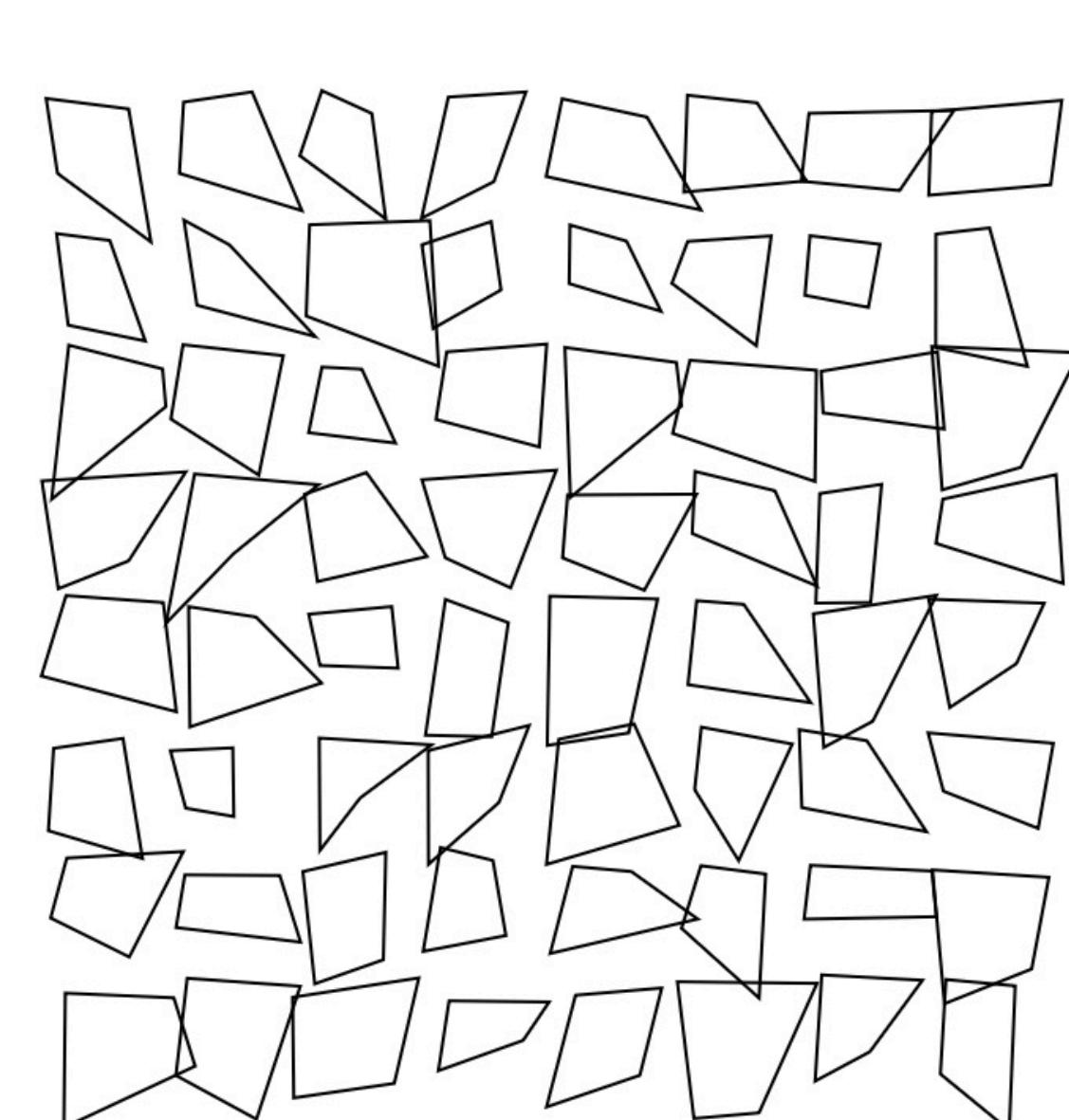


# Tekrar

<https://editor.p5js.org/tbilgin/sketches/YfPEAM7HG>

> sketch.js      Saved: 2 minutes ago      Preview

```
1  function setup() {
2    createCanvas(400,400); // kanvasin boyutu
3    noLoop(); // kod tekrar calismasın
4    noFill(); // üst üste gelen sekiller birbirini ortmesin
5  }
6
7  function draw() {
8    background(255);
9
10   let numShapes = 8; // kenar basina sekil sayisi
11   let size = 40; // her dörtgenin boyutu
12
13   for (i=0; i<1; i++) { //kaç dörtgen üst üste gelsin
14     // dikine tekrar et
15   for (y=0; y<numShapes; y++) {
16     // yatayda tekrar et
17   for (x=0; x<numShapes; x++) {
18     push(); // her dörtgen icin asagidaki koordinatlari baslat
19     translate(40 + x*40, 40 + y*40);
20     quad(
21       random(-5,5), random(-5,5),
22       random(15,45), random(-5,5),
23       random(15, 45), random(15,45),
24       random(-5,5), random(15,45)
25     )
26     pop(); // dörtgen koordinatlarini resetle
27   }
28 }
29 }
30 }
```

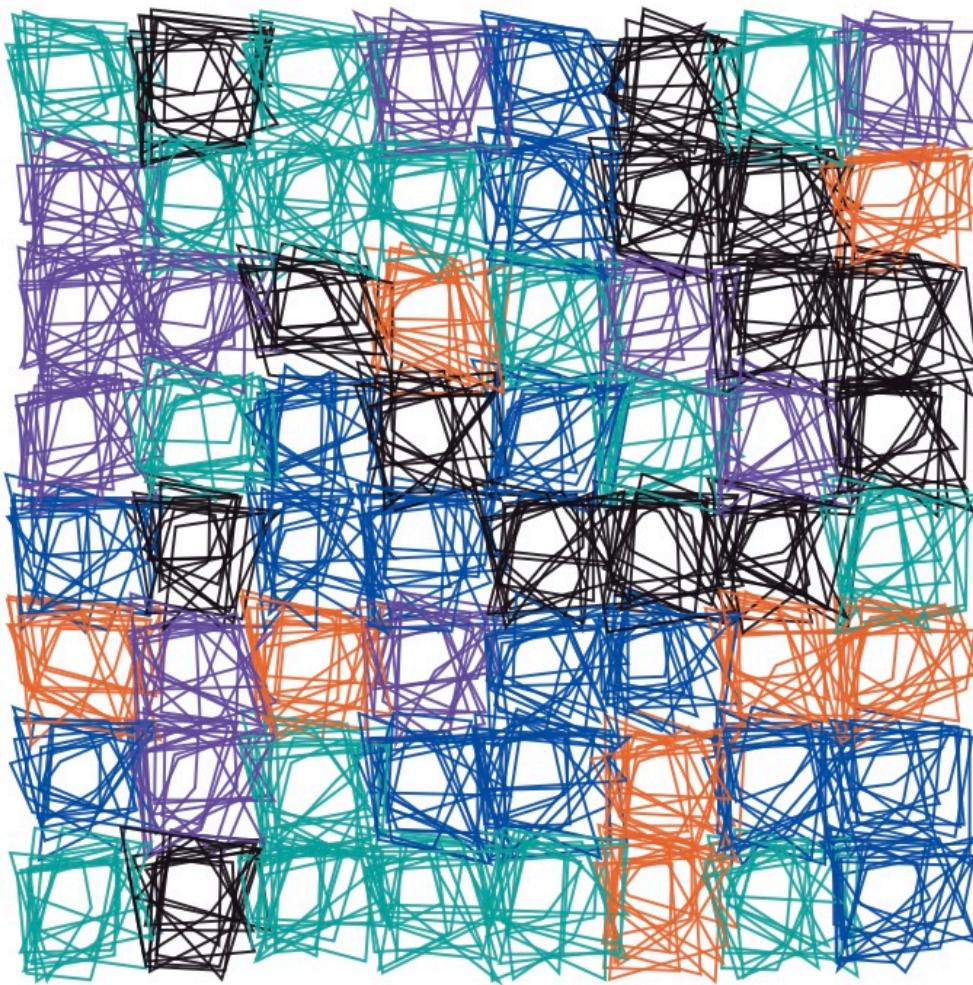


# Renkleri de ekleyelim!

<https://editor.p5js.org/tbilgin/sketches/Li-Bp0Pjc>

> sketch.js • Saved: 32 minutes ago Preview

```
1▼ function setup() {
2  createCanvas(400,400); // kanvasin boyutu
3  noLoop(); // kod tekrar calismasın
4  noFill(); // üst üste gelen sekiller birbirini ortmesin
5  colorMode(HSB); // renk paleti
6 }
7
8▼ function draw() {
9  background(255); // kanvasin rengi
10
11  let numShapes = 8; // kenar basina dörtgen sayisi
12  let size = 40; // her dörtgenin boyutu
13
14  // Renkler
15  let vera = [color(18, 80, 91, 0.9), color(178, 100, 63, 0.9),
16  color(212, 100, 63, 0.9), color(300, 29, 9, 0.9), color(257, 52, 61,
17  0.9)]; // dört renk
18  let shapeColors = [];
19  for (c=0; c<(numShapes*numShapes); c++) {
20    shapeColors.push(random(vera)); // renklerin kombinasyonu
21
22  for (i=0; i<12; i++) { // kaç dörtgen üst üste gelsin: katman
23    let s=0; // kaçinci katmandayız
24    // dikine dörtgen ekle
25    for (y=0; y<numShapes; y++) {
26      // enine dörtgen ekle
27      for (x=0; x<numShapes; x++) {
28        push(); // her dörtgen için aşağıdaki koordinatları başlat
29        translate(40 + x*40, 40 + y*40); // dörtgeni kaydır
30        stroke(shapeColors[s]) // dörtgenin rengi - her katmanda aynı
31        quad(
32          random(-10,10), random(-10,10),
33          random(15,45), random(-5,5),
34          random(15, 45), random(15,45),
35          random(-5,5), random(15,45)
36        )
37        pop(); // dörtgen koordinatlarını tekrar sıfırla
38      }
39    }
40  }
41}
```



## Kendi dijital eserimiz nasıl olmuş?

<https://padlet.com/bilg1/pera-da-yaratıcı-kodlama-at-lyesi-lhbe45t4bf3lit4c>

