

# storytelling with data

Visualizing data  
with code

- hello!
- deep dive: data & storytelling
- creative coding

goal: 3h–4h

⌚ 1h (?)

## 😊 break (20 mins)

- your project

⌚ 5-10 mins / team

## 😊 break?

- let's code

⌚ rest of time

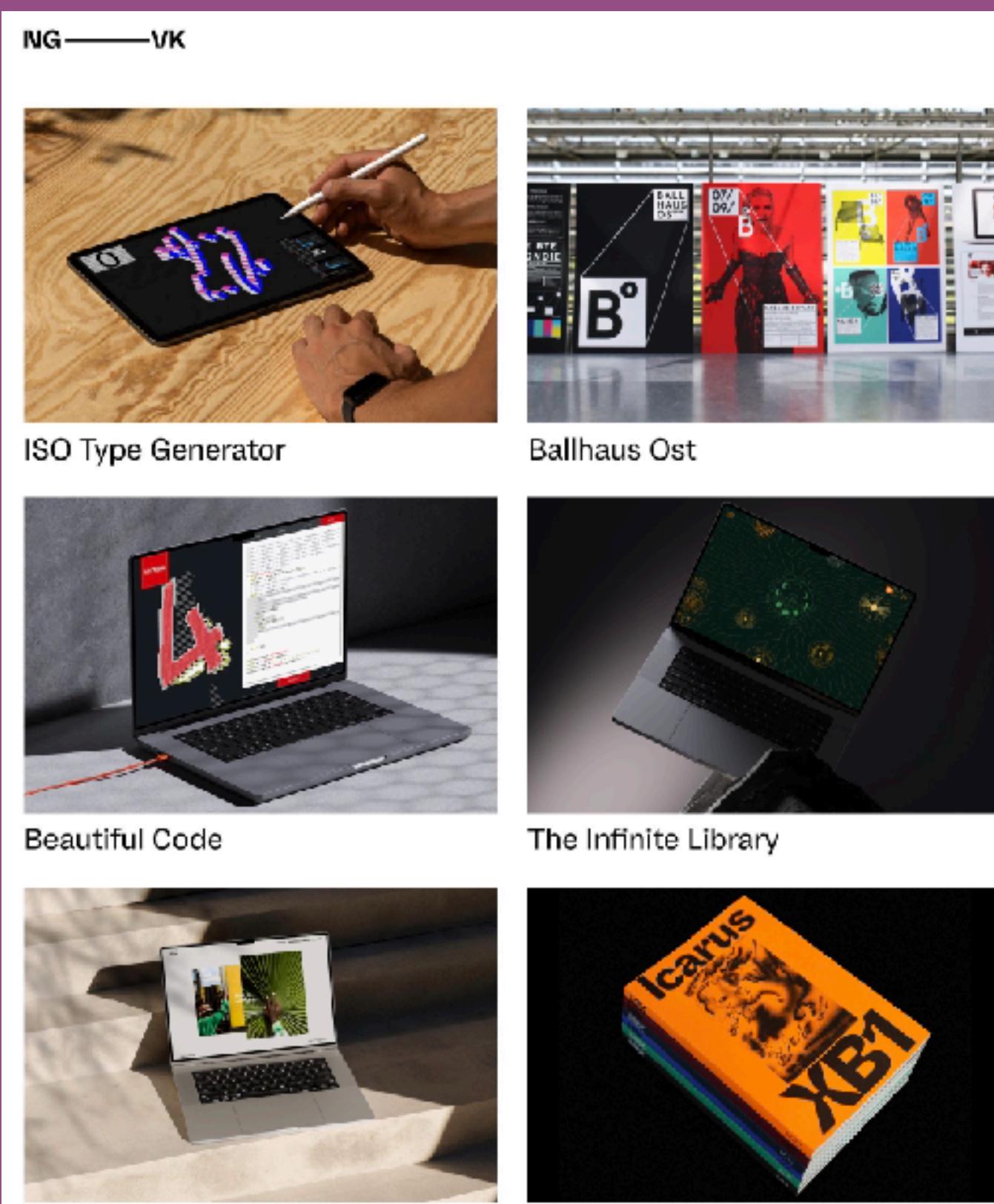


Hello!

# Prague visual storytelling design, code & web science communication creative coding



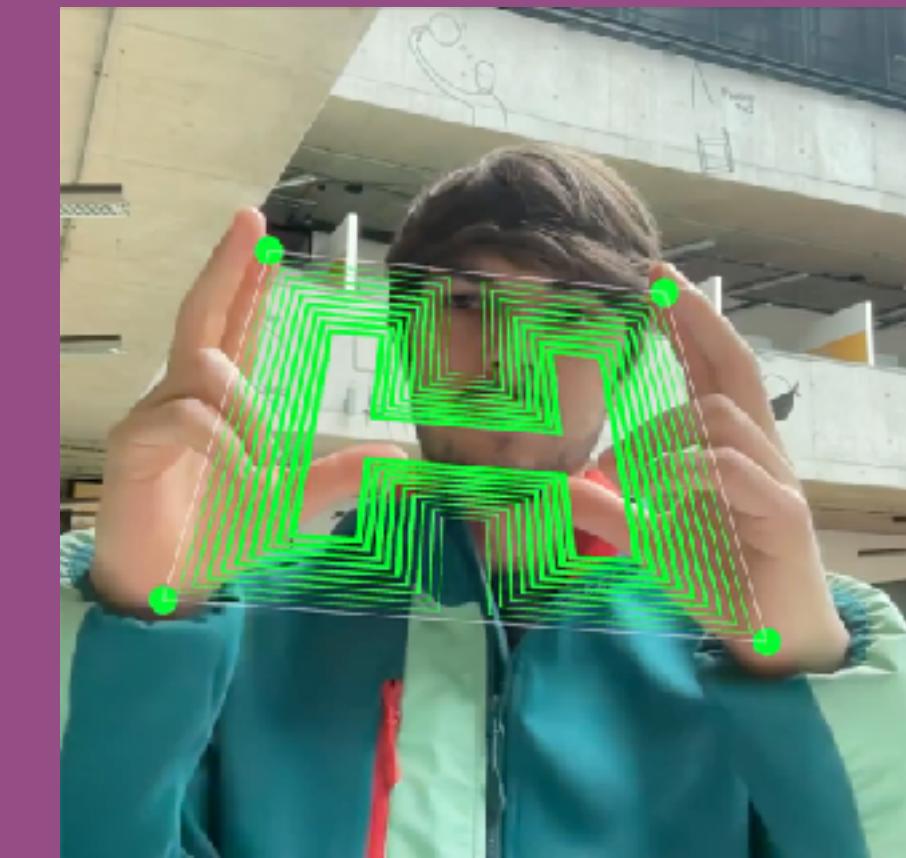
## Webdesign, Branding, Frontend



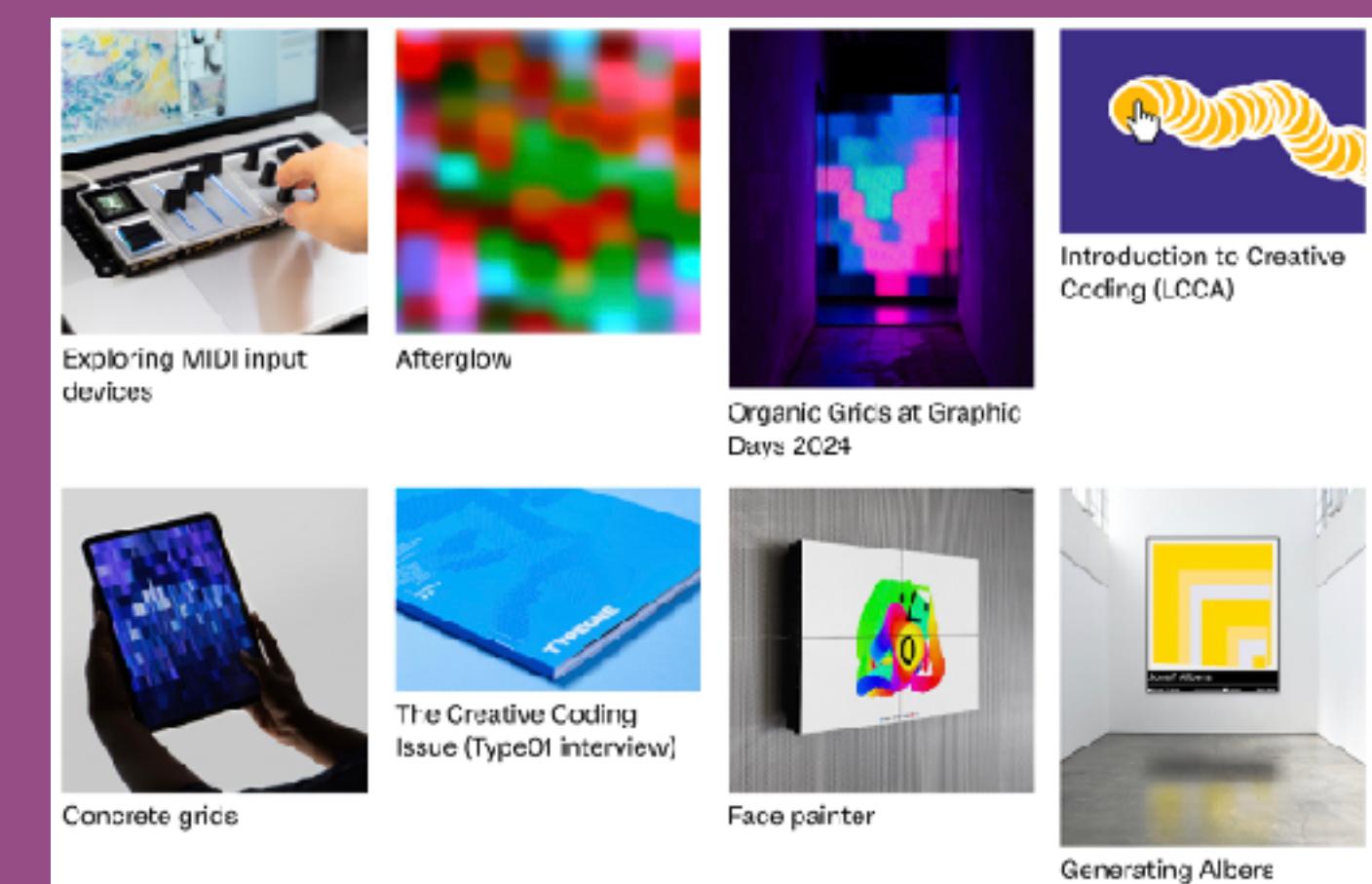
<https://nahuelgerth.de/>



## Embodied Interaction



## Generative Tools

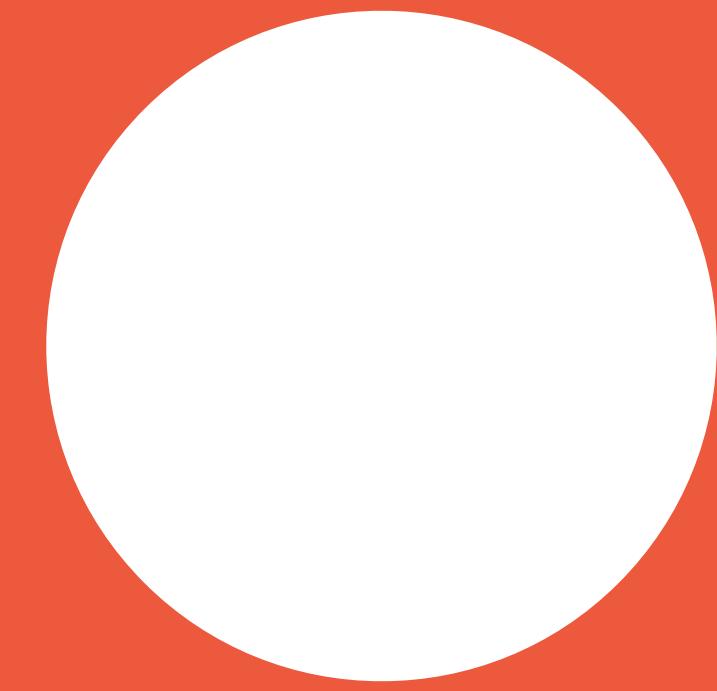




science

art

tech



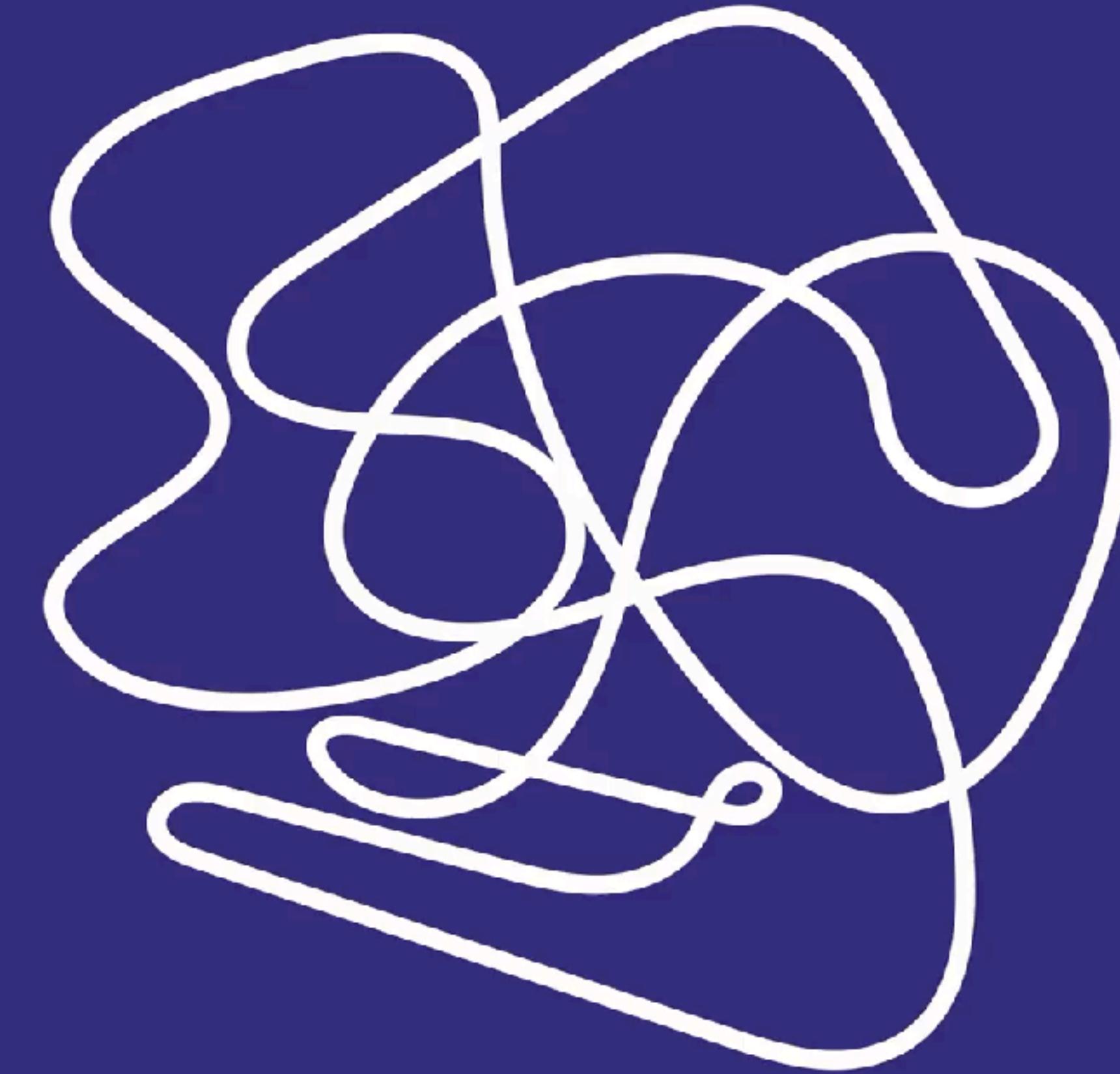
data and  
storytelling

Data is everywhere.



We are flooded with data every day.

# How can we make sense of this mess?

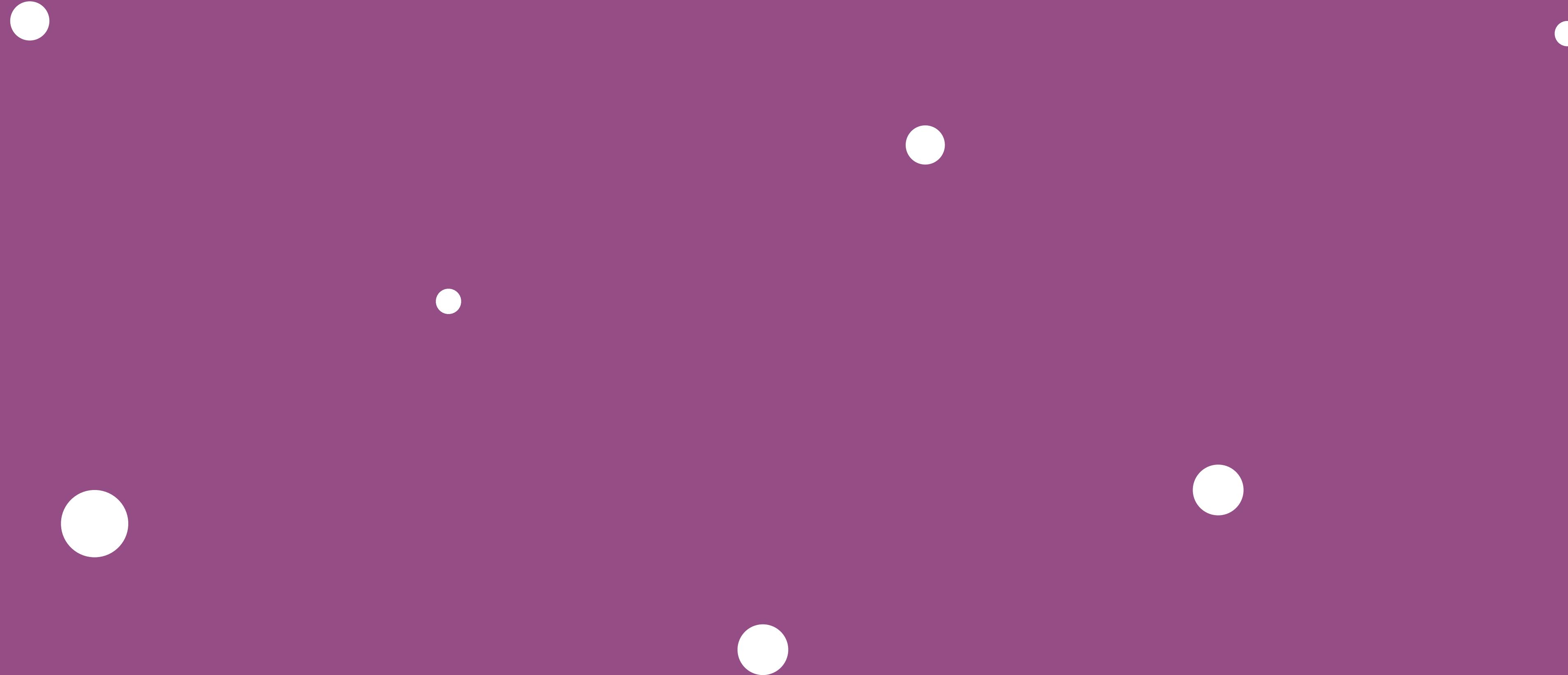


Let's take a step back ...

# Humans are storytelling creatures.



Data itself is meaningless to us ...



... we respond to emotion, not data.

We need a personal connection to data.



You

Why should it matter to us?

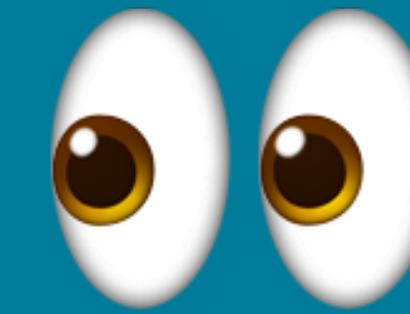
How is it connected to us?

A story connects pieces of data and makes sense of it.



Suddenly we have framework.

Through stories we can evoke emotion.



We can make data relatable and connect to shared experiences.

# We expect stories in a certain structure.



## 1. set up

Where does your story begin?

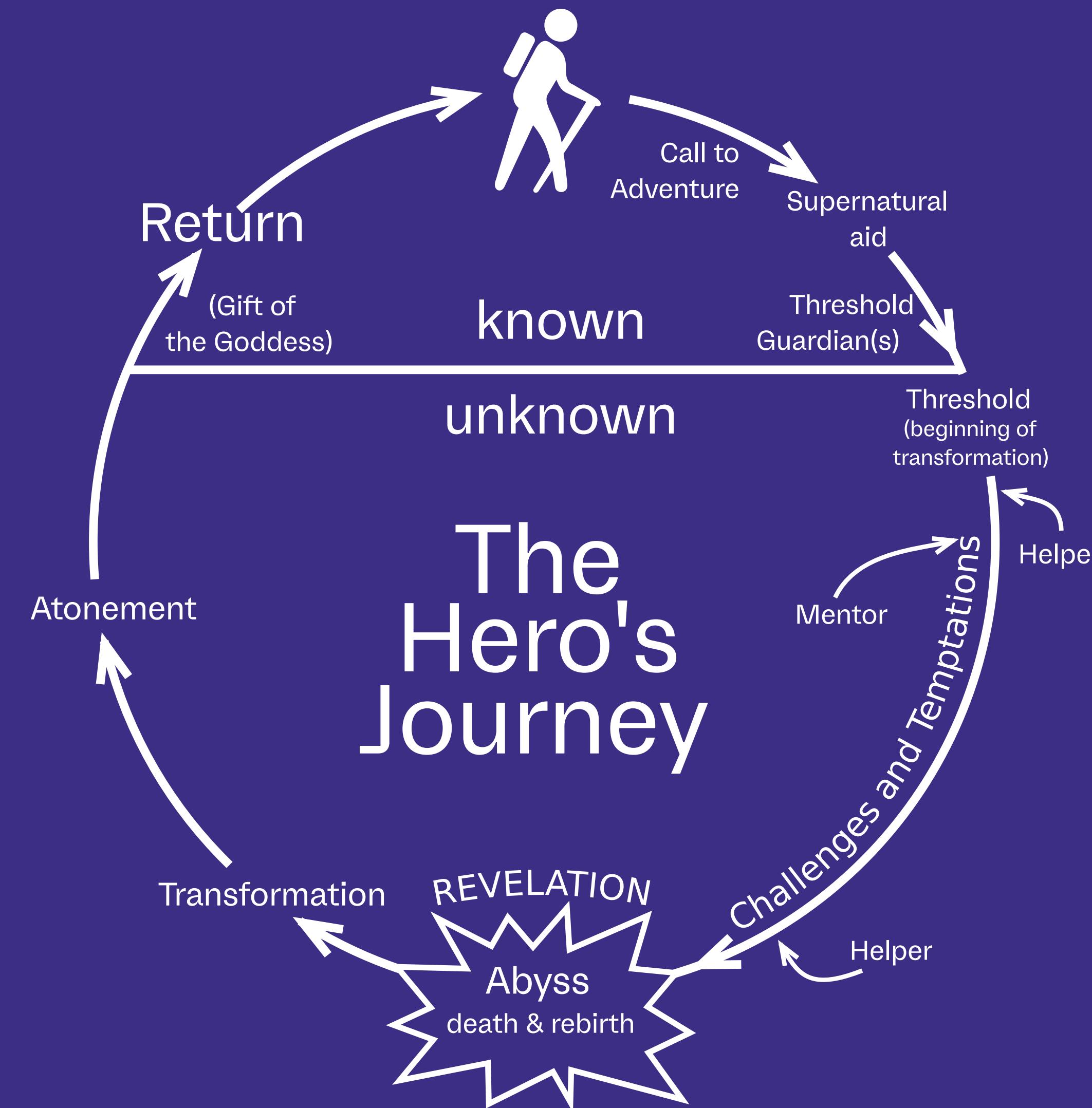
## 2. confrontation

What problem or question  
do you address?

## 3. resolution

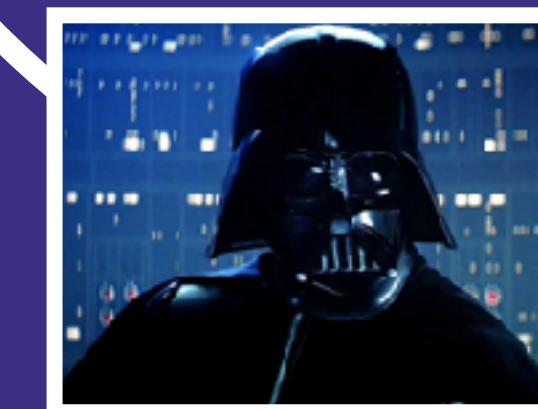
What insight do you  
create with this?

# We can analyze these patterns in detail.



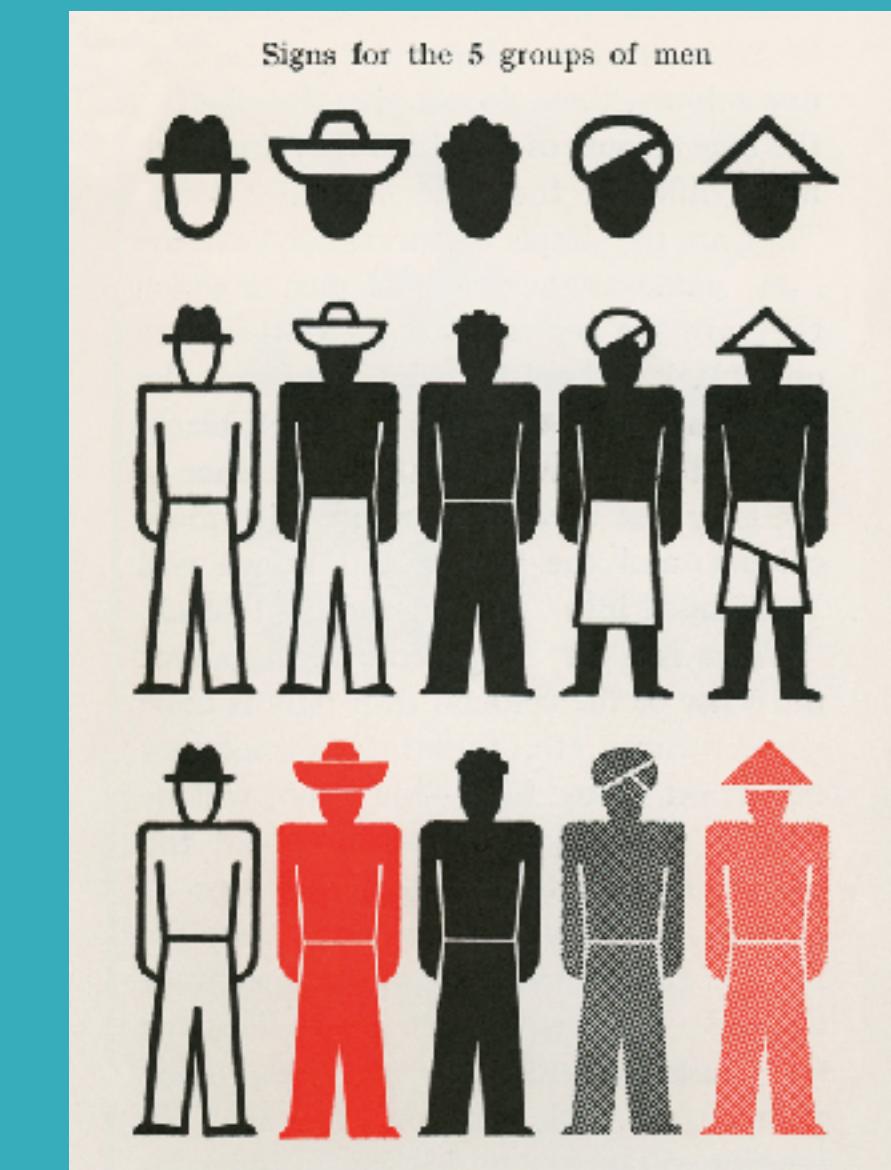
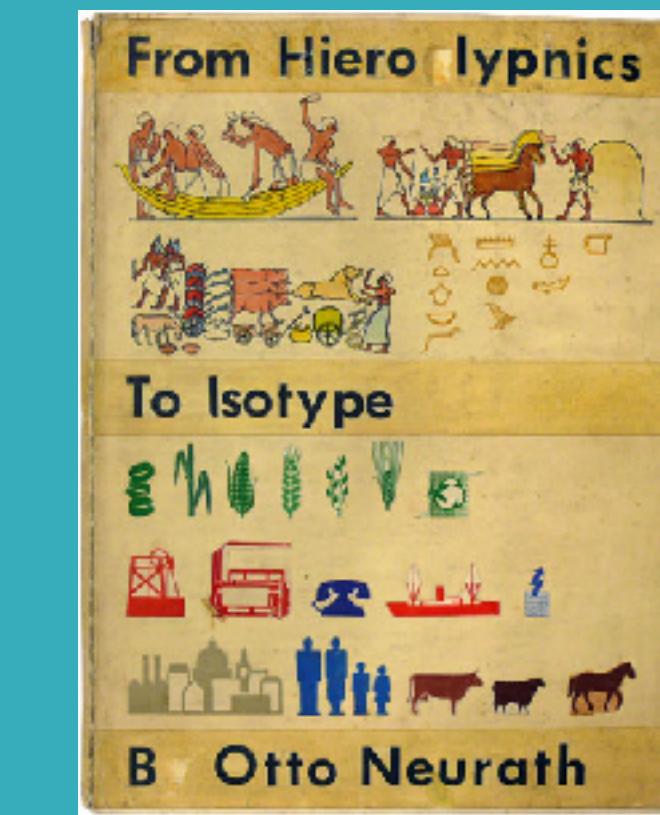
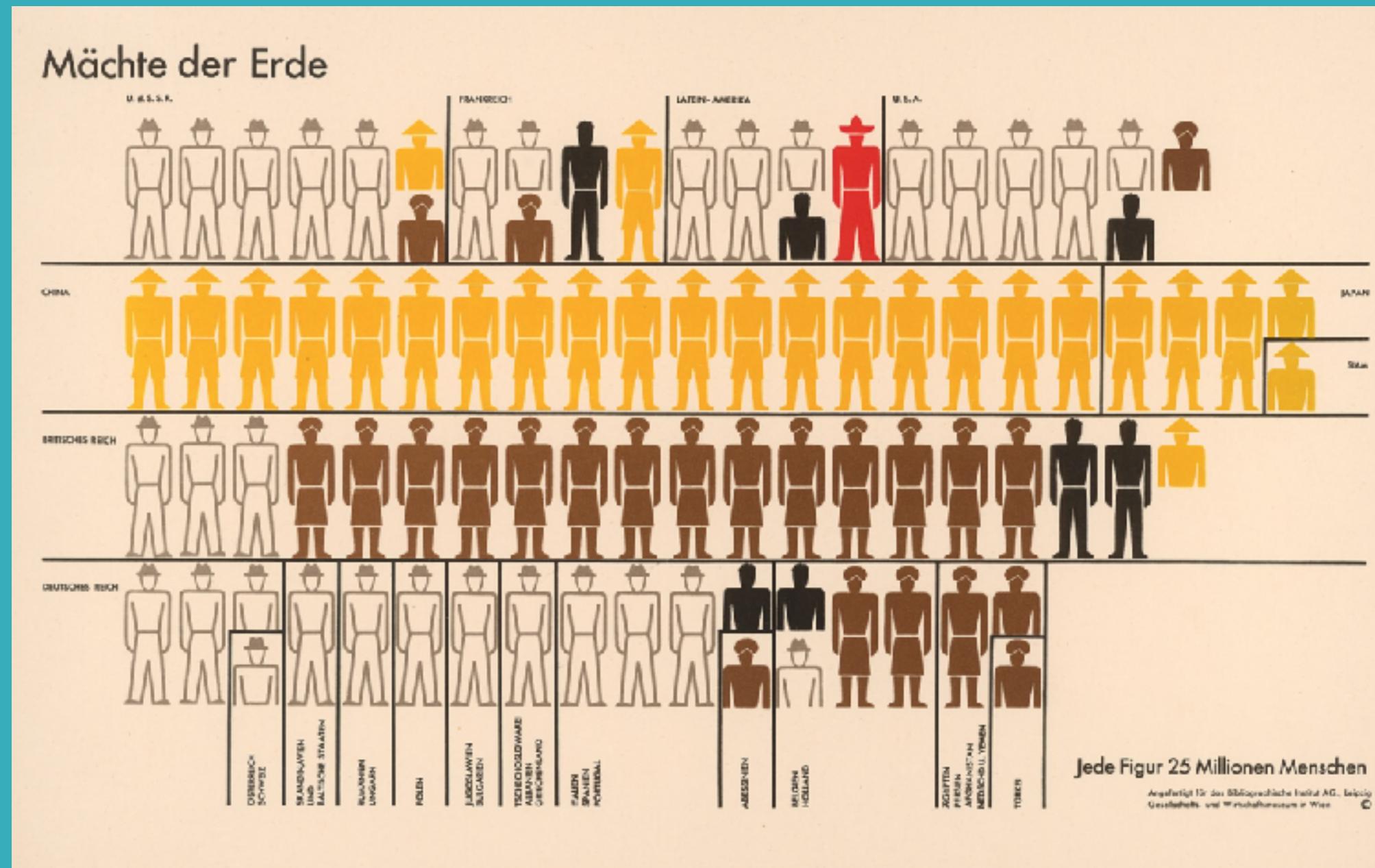
**side task:**  
analyze your favorite  
movies/stories for their  
narrative structure

We can analyze these patterns in detail.



# As designers we can make use of those learned patterns and apply them visually.

Otto Neurath, Isotype, 1925 –1934



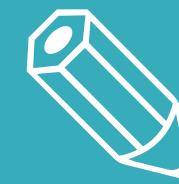
# AIDA

A – attention

I – interest

D – desire

A – action



## Find your story foundation



Data

What stands out to you?



Audience

Who are you speaking to?



What is the core message?



Perspective

From which point of view?



Find metaphors

(visual or conceptual)

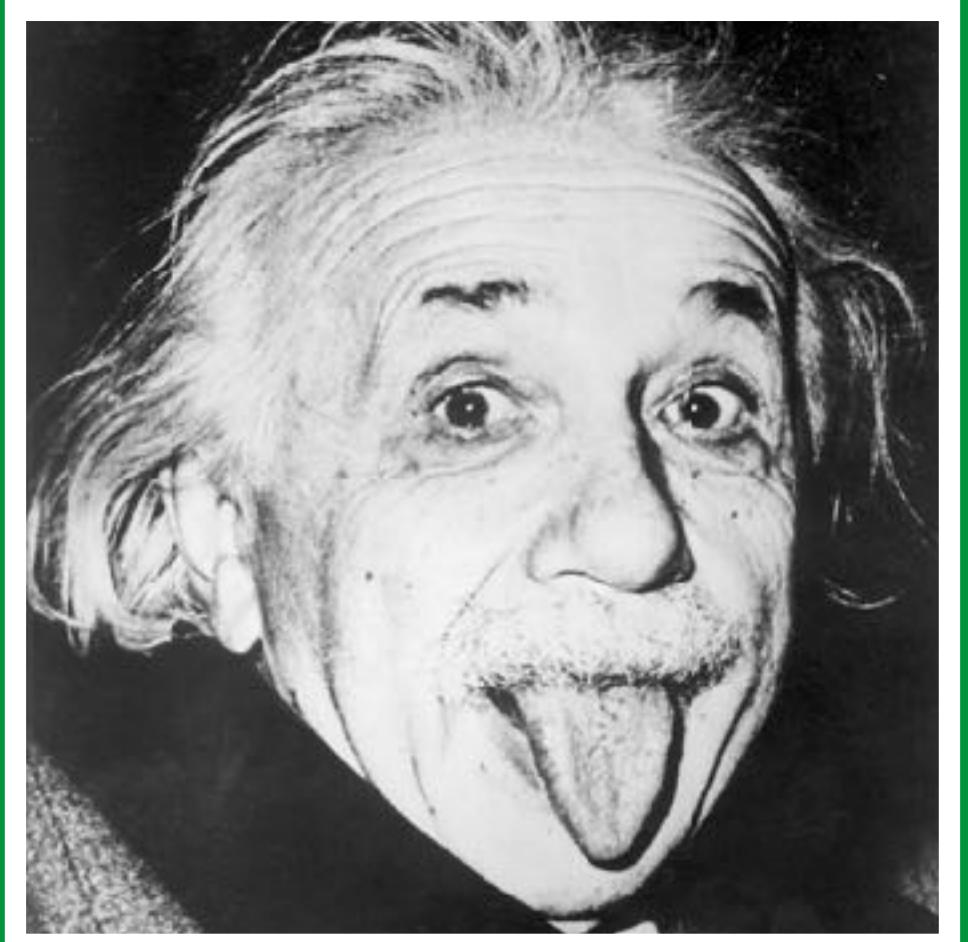


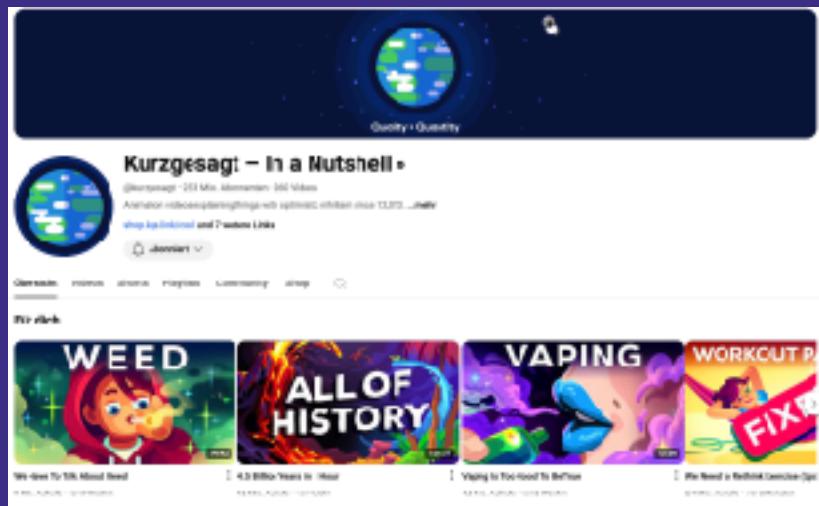
Use emotion

→ treat data as  
a raw material:  
  
filter it,  
evaluate it,  
embed it into context.

If you can't explain it simply,  
you don't understand it  
well enough.

Albert Einstein





## Kurzgesagt

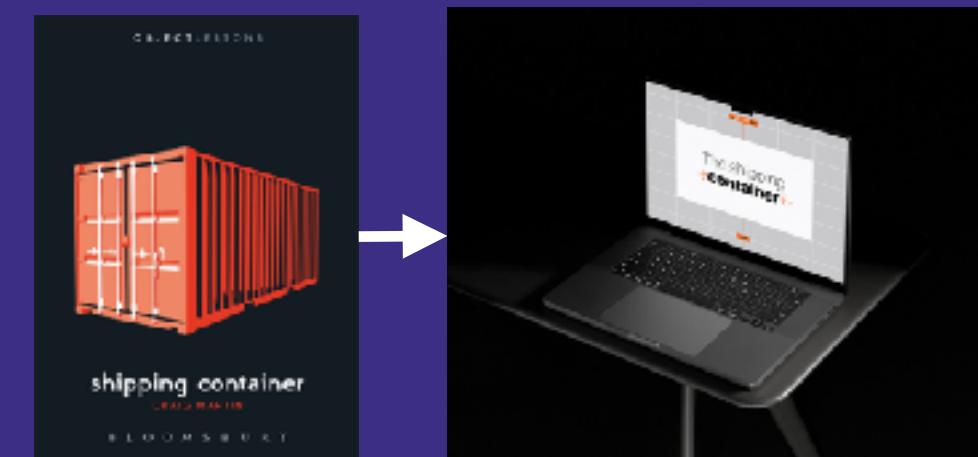
<https://www.youtube.com/@kurzgesagt>



## Big History

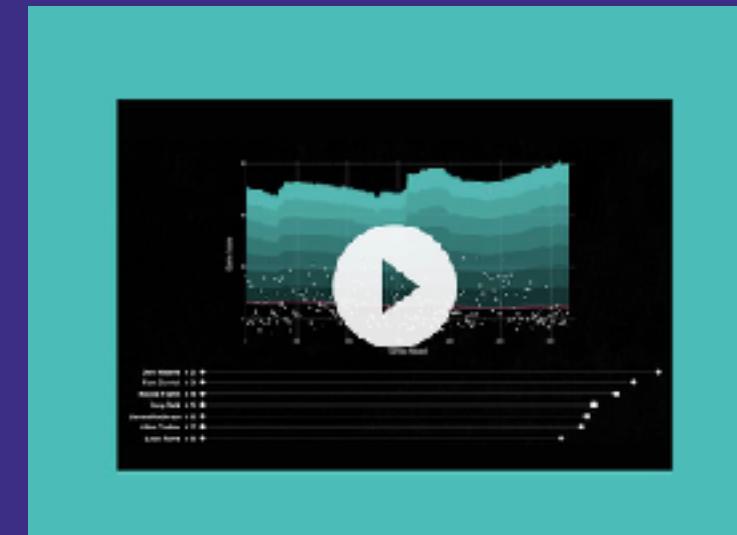
History of the universe in 18 minutes

[https://www.ted.com/talks/david\\_christian\\_the\\_history\\_of\\_our\\_world\\_in\\_18\\_minutes?subtitle=en&lng=de](https://www.ted.com/talks/david_christian_the_history_of_our_world_in_18_minutes?subtitle=en&lng=de)



## Object lessons & Shipping container

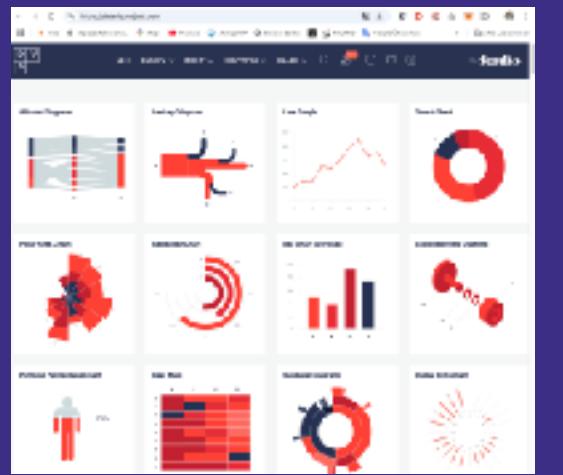
<https://containerization.info/>



## The Pudding

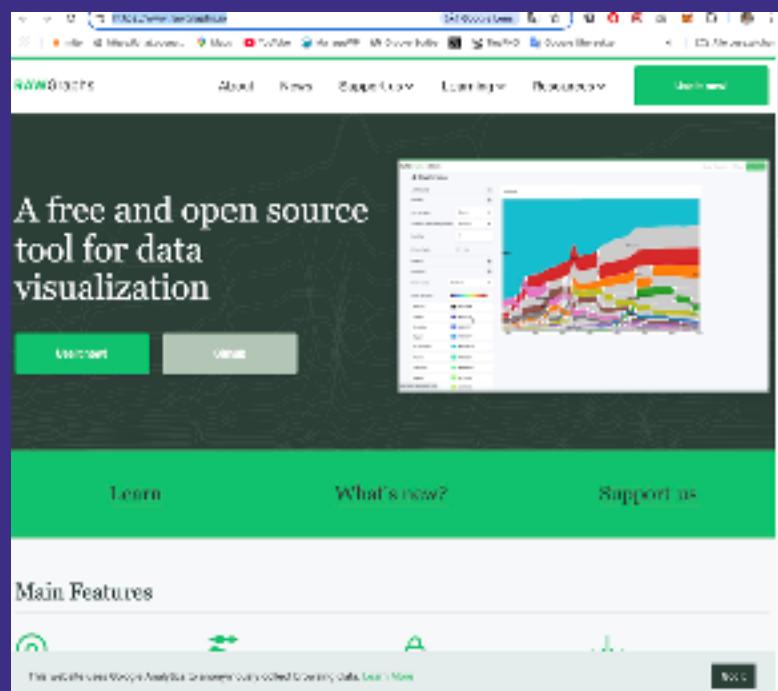
Looooots of visual data essays

<https://pudding.cool/>



## Data Viz Project

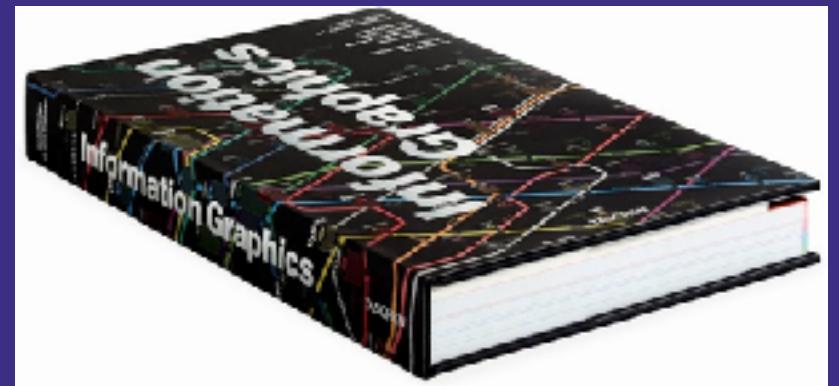
<https://datavizproject.com/>



## Rawgraphs

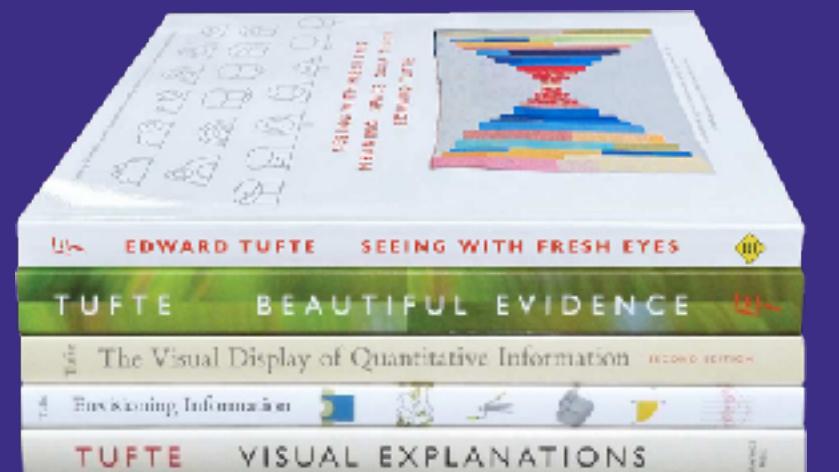
An open source tool for creating info graphics

<https://www.rawgraphs.io/>



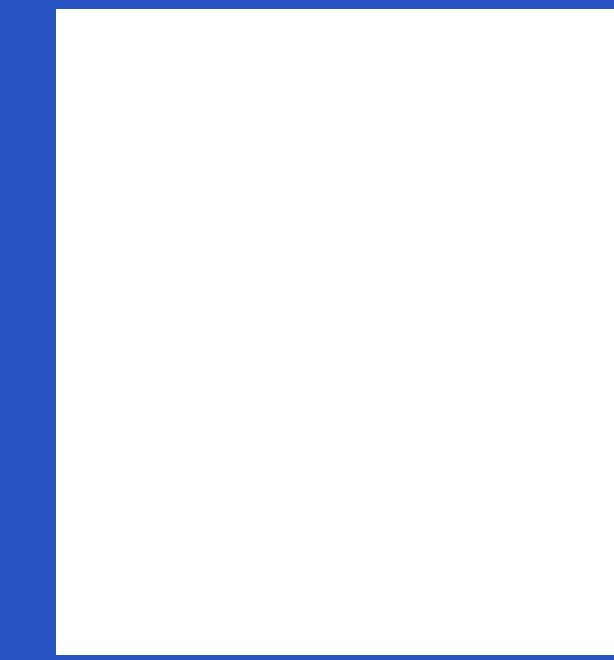
## Information Graphics

<https://www.taschen.com/en/books/graphic-design/44653/information-graphics>



## Edward Tufte

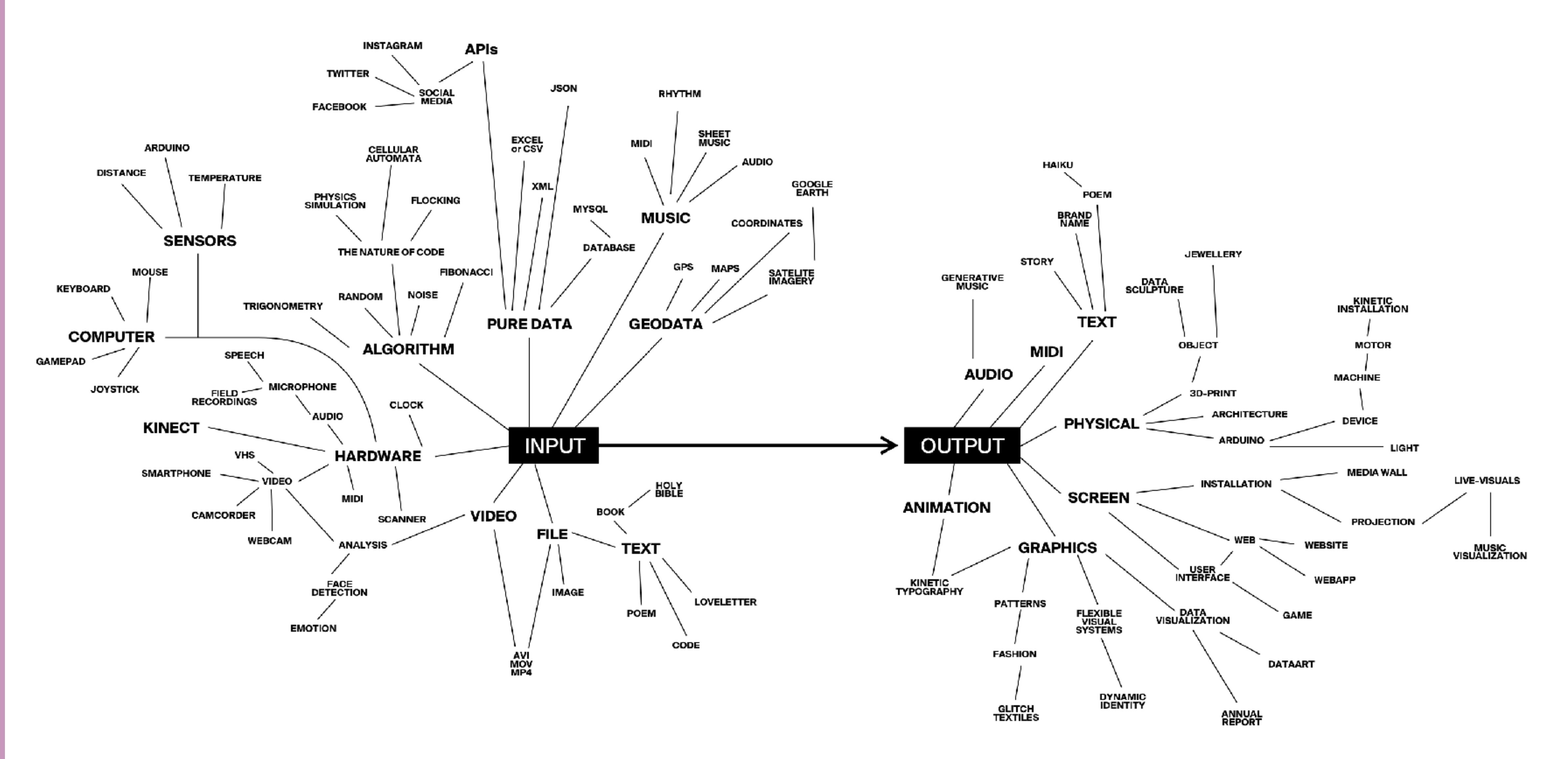
Science of data visualization.



creative  
coding

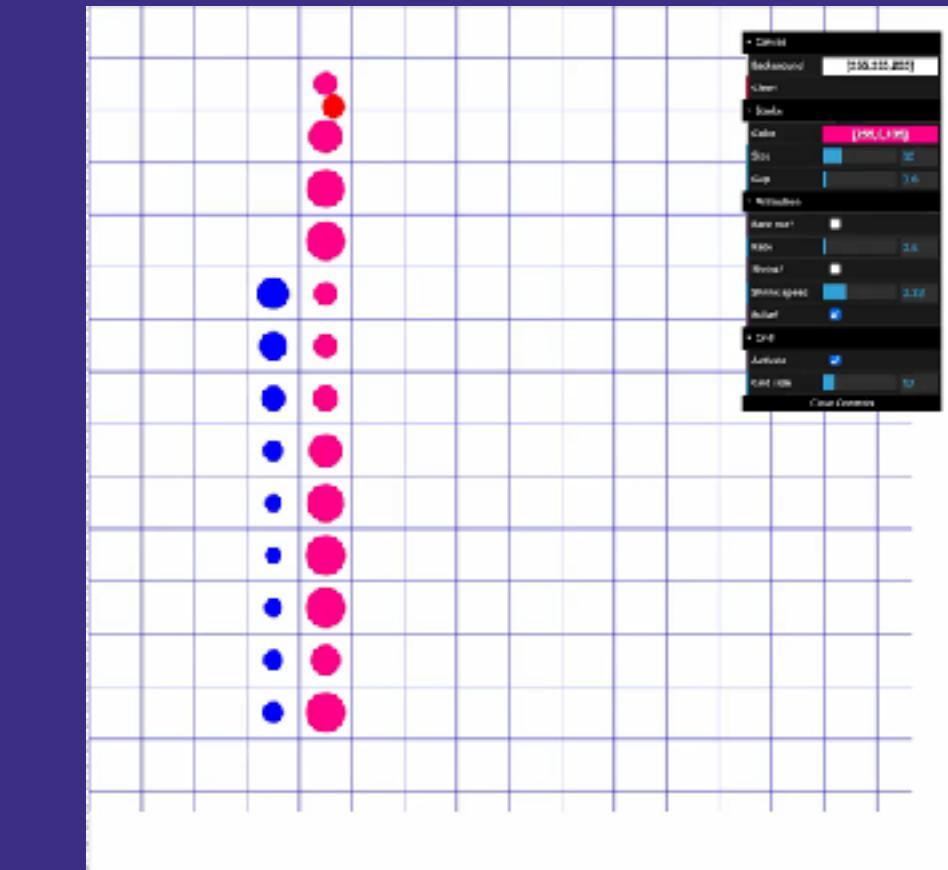
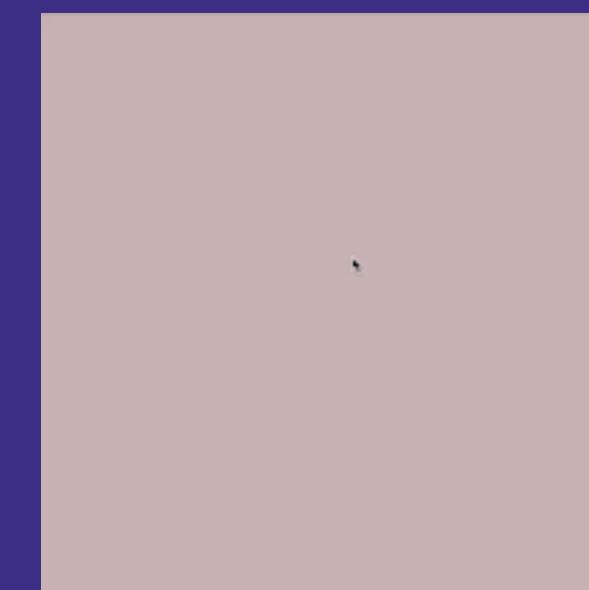
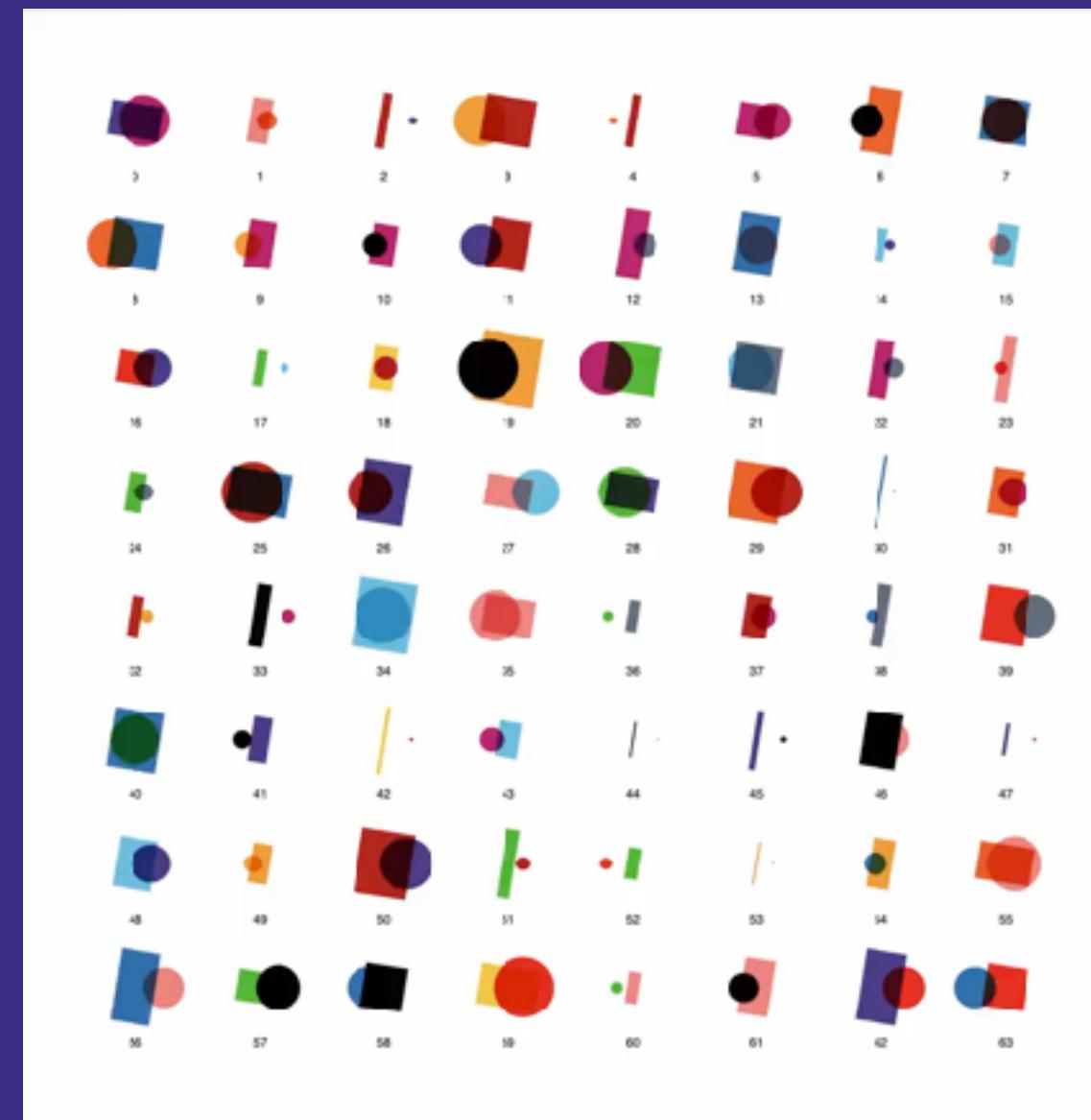
# Creative coding with p5.js

- p5.js is the tool we use to analyze and visualize our data
- it is a web-based coding language (basically we will create a mini-website)
- you can create all sorts of crazy stuff with it:
  - interactive / animated
  - series comparing datasets
  - random compositions



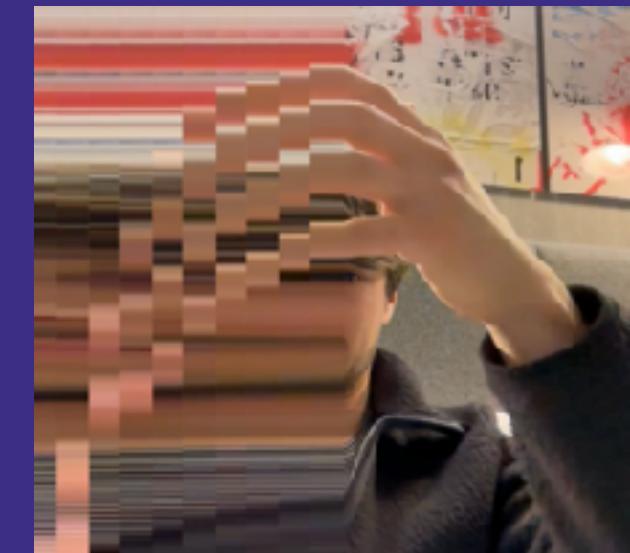
We can code ...

... motion



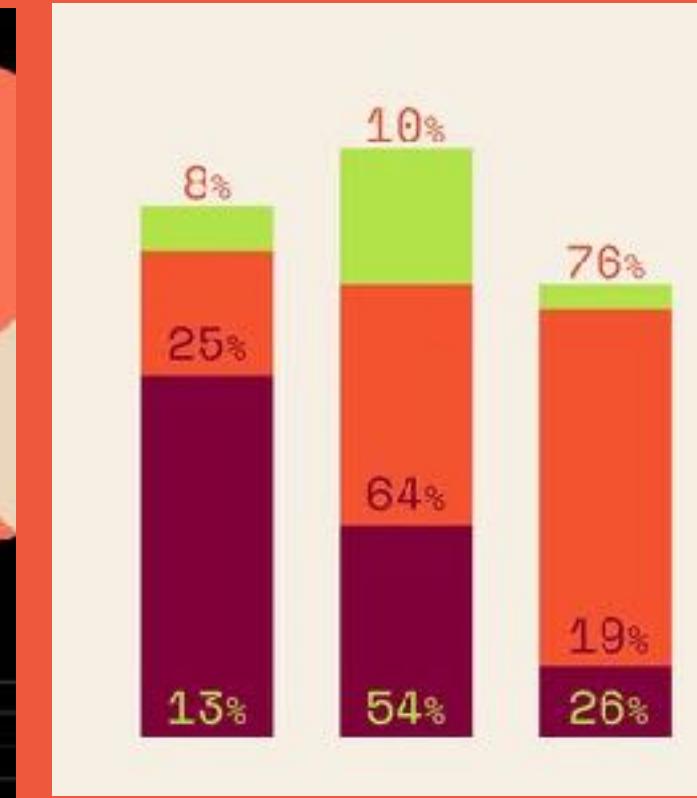
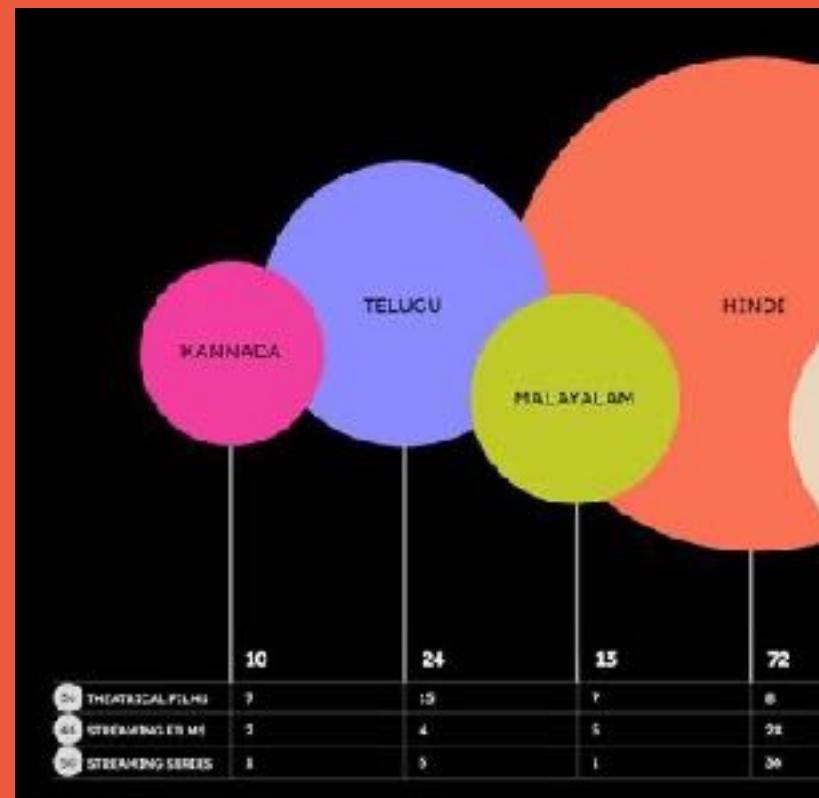
... design

... interaction

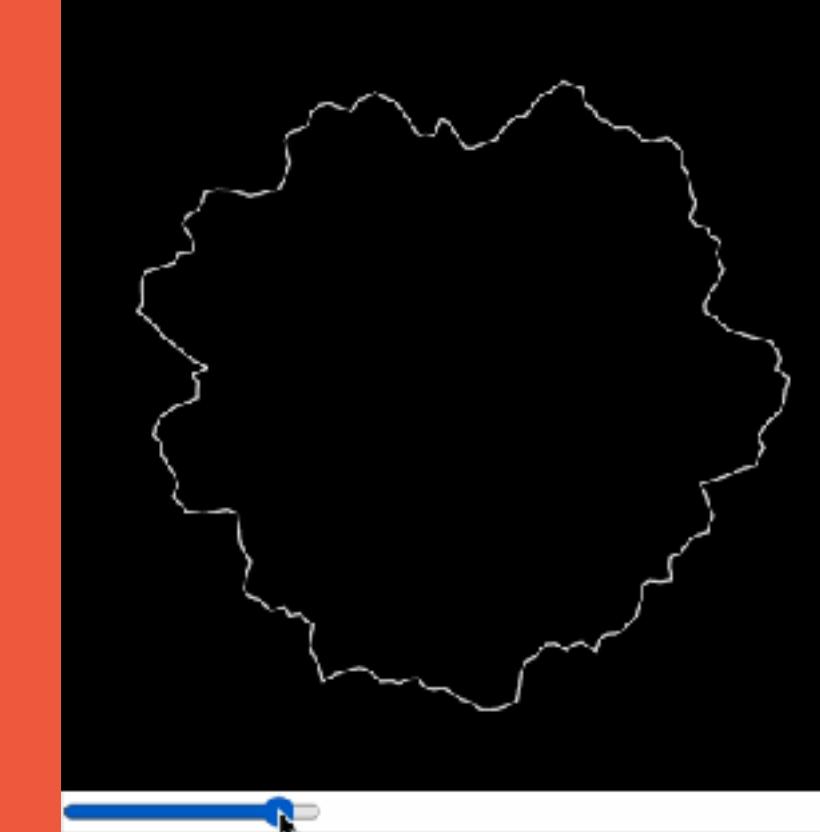


# Things we could look into

interpreting .csv tables



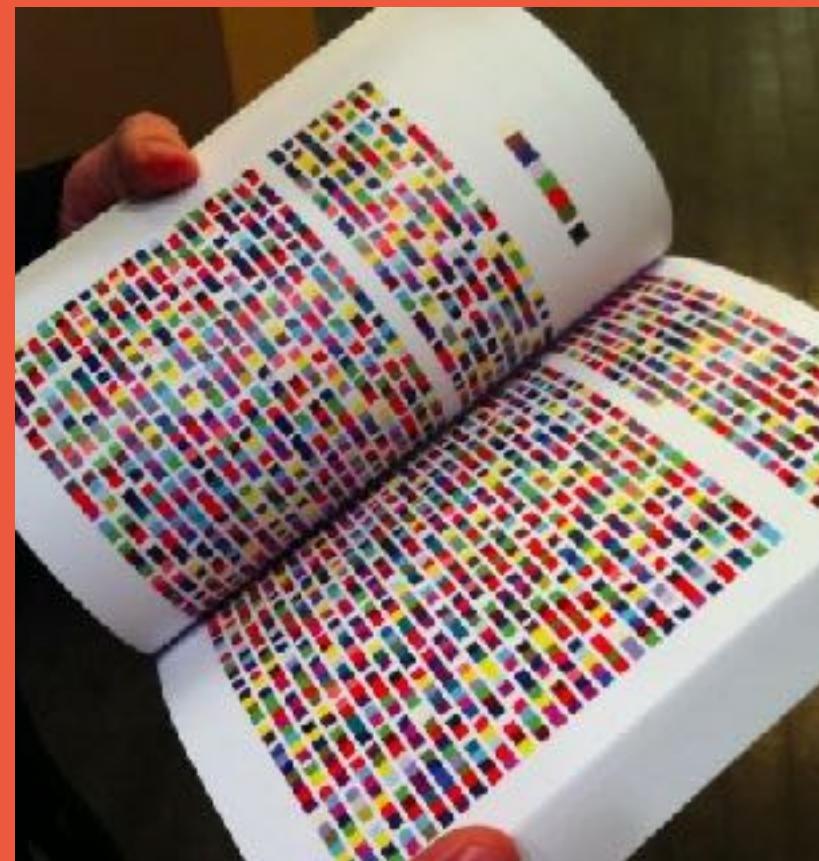
visualizing sound



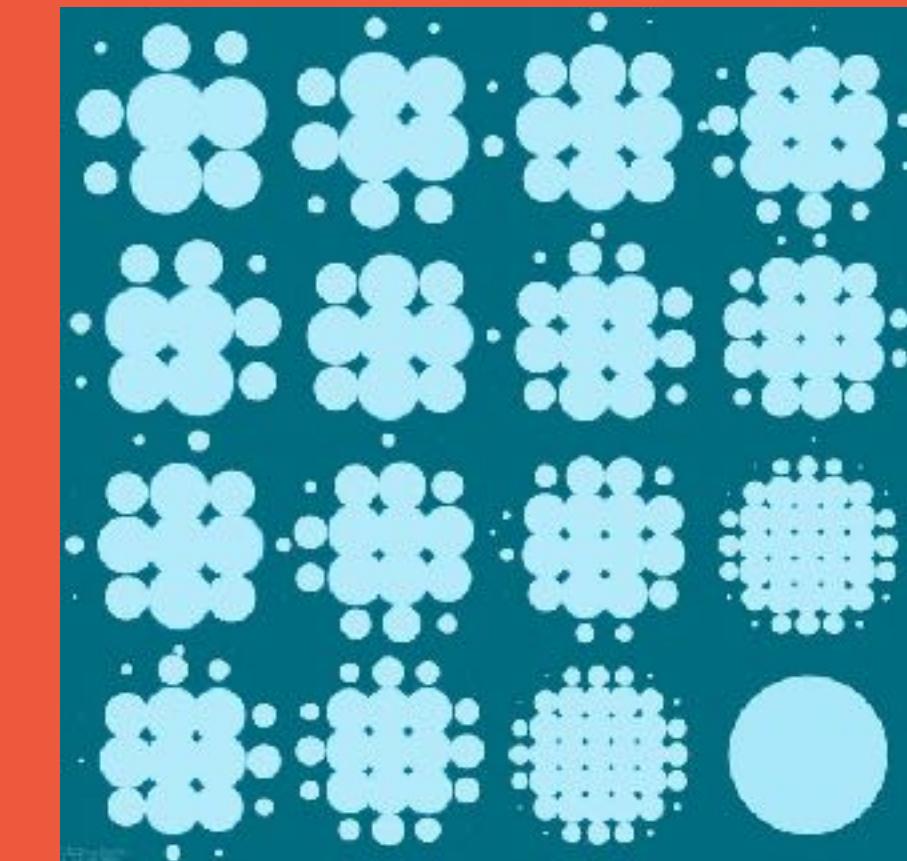
interpreting pixels



evaluating text



series with randomized values



# Brands in motion – North Sea Jazz



- AGENCY  
Studio Dumbar  
[studiodumbar.com](http://studiodumbar.com)
- INPUT  
letters
- OUTPUT  
branding , dynamic motion system,  
corporate visual language, posters



# Design generators – P5 studio



- ARTIST  
Tim Rodenbröker  
[timrodenbroeker.de](http://timrodenbroeker.de)
- INPUT  
values, text, image, video
- OUTPUT  
poster generator, web experience

**p5studio**

HEADLINE	HIDE
SUBLINE	HIDE
IMAGE	HIDE
IMAGE2	HIDE
GRID	HIDDEN
BASE	HIDE

TEXT

WORKSHOP  
30.11.19  
FH BIELEFELD  
FB GESTALTUNG

FONTSIZE

LINEHEIGHT

FONT Roboto-Medium.ttf

ALIGN == == =



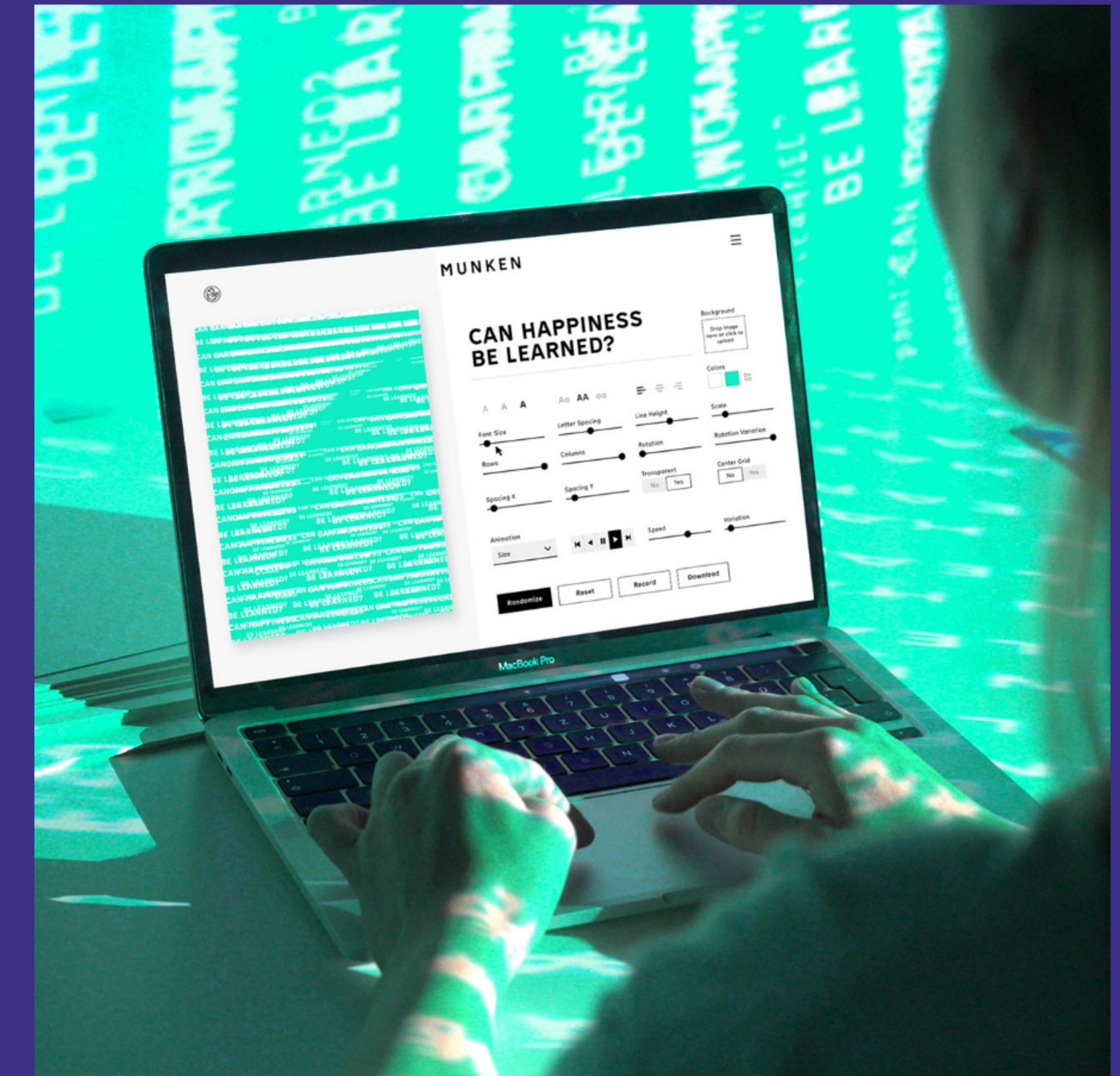
# Brands in motion – squarespace

- AGENCY  
DIA Studio  
dia.tv
  - INPUT  
letters, logo, shapes
  - OUTPUT  
branding , dynamic motion system,  
corporate visual language



# Design generators – munken creator

- AGENCY  
Patrik Hübner, JUNO  
[patrik-huebner.com](http://patrik-huebner.com)
- INPUT  
custom values, text, image, video
- OUTPUT  
motion design, web experience,  
asset generator



# Learning resources (p5.js)

Tim Rodenbröker

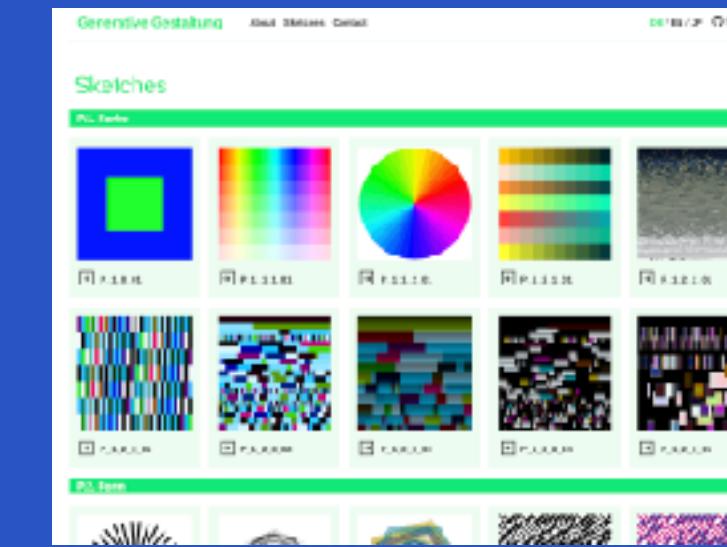
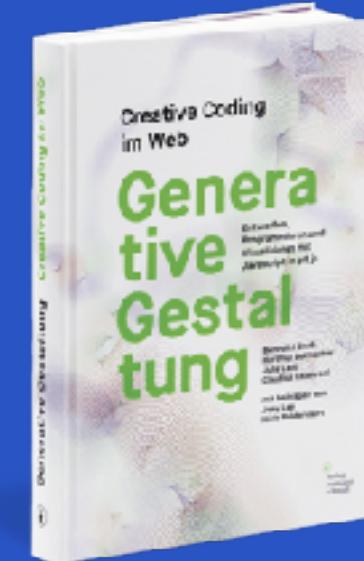


Hi, my name is Tim  
and I build an  
online school for  
Creative Coding in  
Design.  
[Learn more](#) [Join](#)

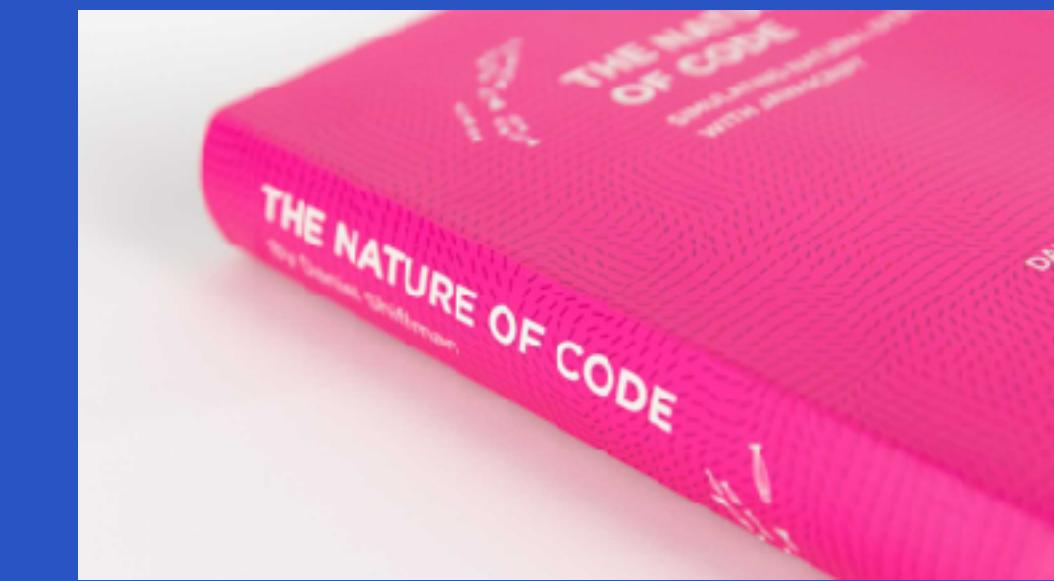
[blog/latest →](#)

<https://timrodenbroeker.de/>

Generative  
Gestaltung

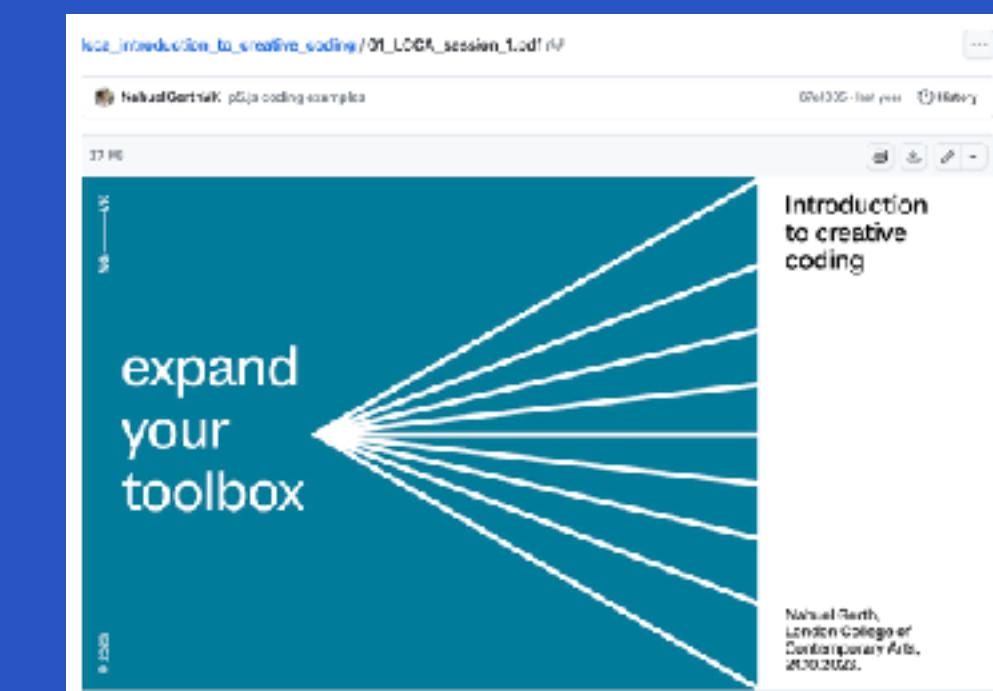


<http://www.generative-gestaltung.de/2/>



Coding Train &  
Nature of Code

<https://natureofcode.com/>  
<https://thecodingtrain.com/>



[https://github.com/NahuelGerthVK/  
lcca\\_introduction\\_to\\_creative\\_coding](https://github.com/NahuelGerthVK/lcca_introduction_to_creative_coding)

My courses  
on GitHub

## Zach Lieberman

[instagram.com/zach.lieberman](https://instagram.com/zach.lieberman)  
daily code experiments

## Tim Rodenbröker

[timrodenbroeker.de](https://timrodenbroeker.de)  
onboarding courses

## Studio Dumbar

[studiodumbar.com](https://studiodumbar.com)  
branding and motion design

## Vera van de Seyp

[instagram.com/veravandeseyp](https://instagram.com/veravandeseyp)  
crazy type experiments

## The Coding Train

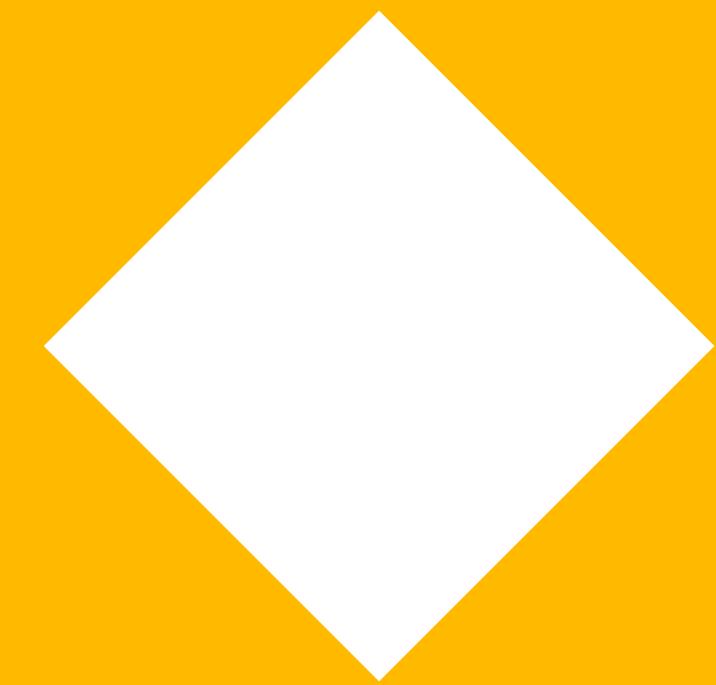
[youtube.com/@TheCodingTrain](https://youtube.com/@TheCodingTrain)  
endless tutorials

## Dia.TV

[dia.tv](https://dia.tv)  
branding and motion design

# 20 mins break

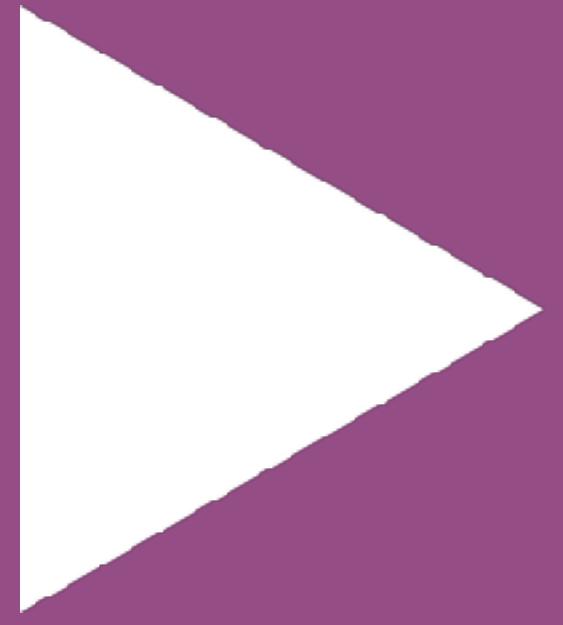




your  
project

# 35 years under the rule of money

- tell me about your project ideas
- what kind of data will you work with?
- 5-10 mins / team
- goal:  
I need a clear idea which examples to prepare for our next session :-)



a little bit  
of coding

# setting up our workspace

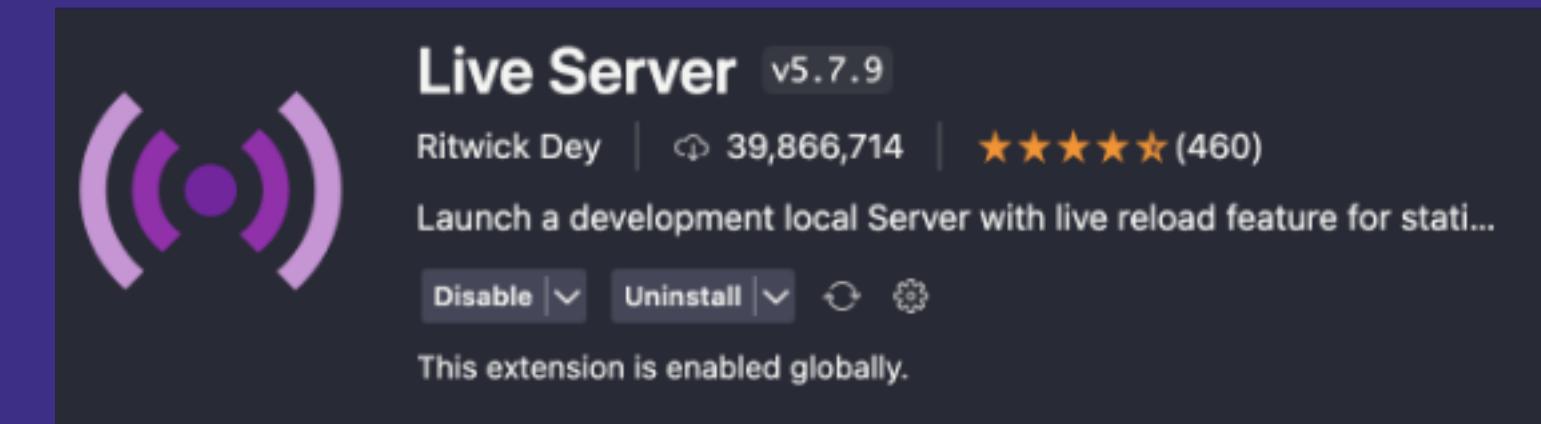


install VS code  
our code editor

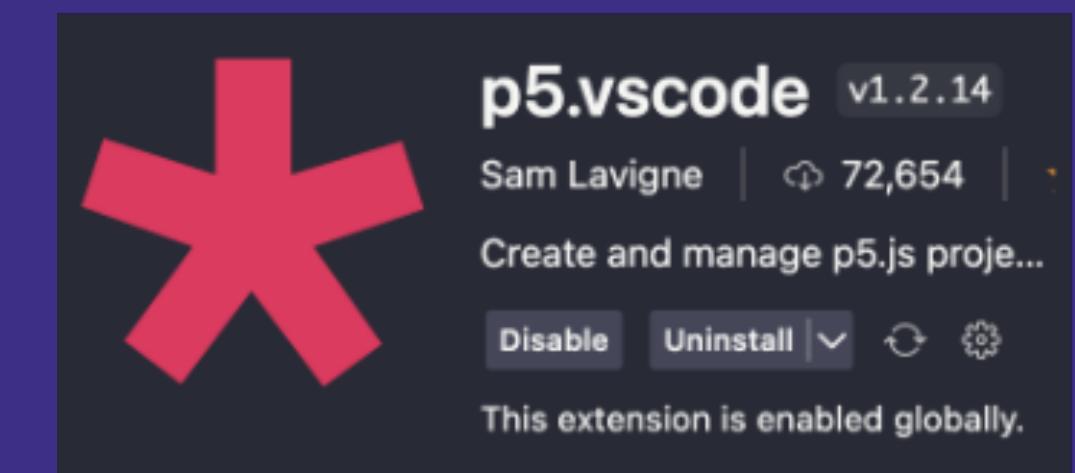


we use two extensions  
VS code/view/extensions

## Live Server (Ritwick Dey)

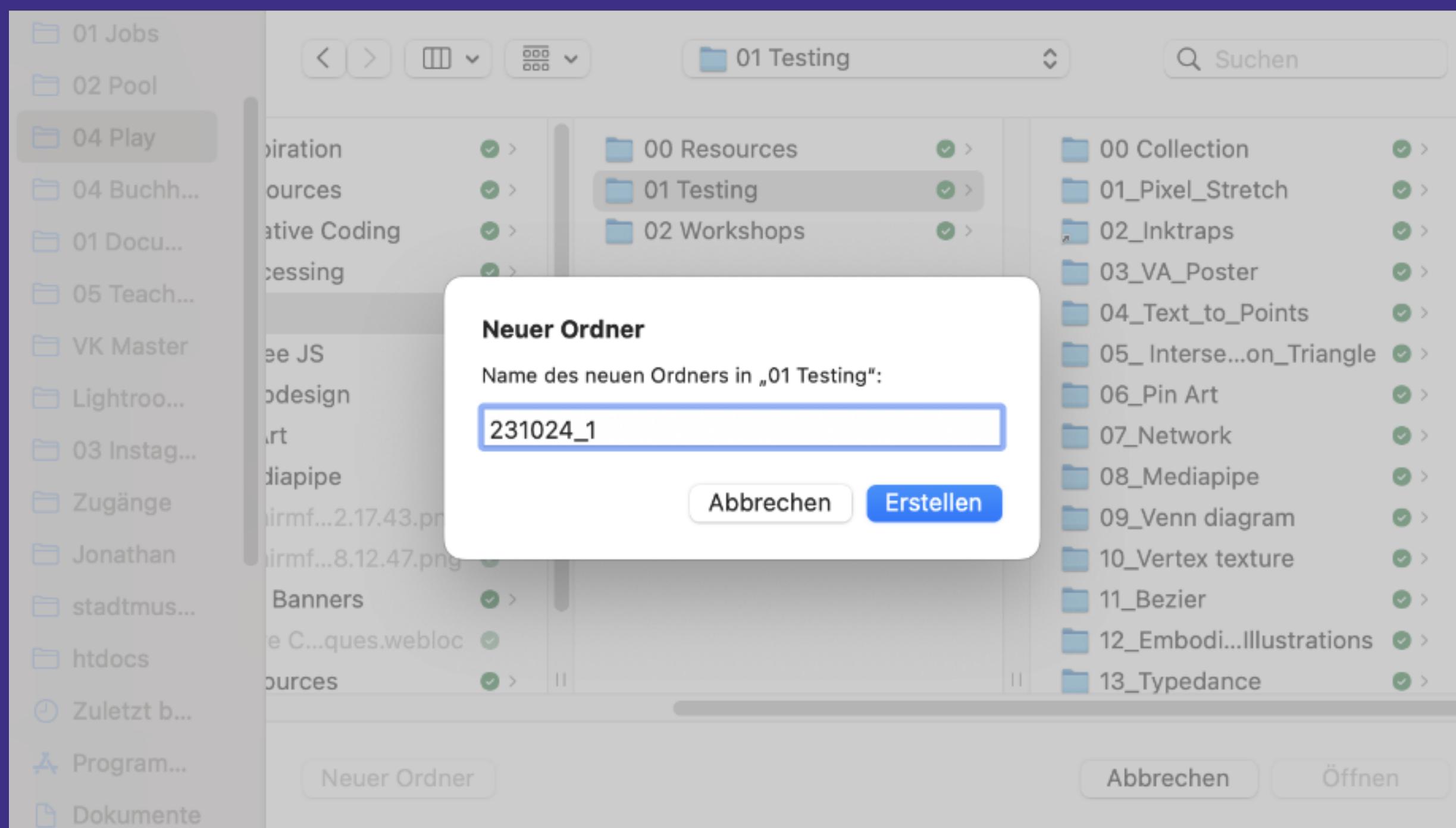


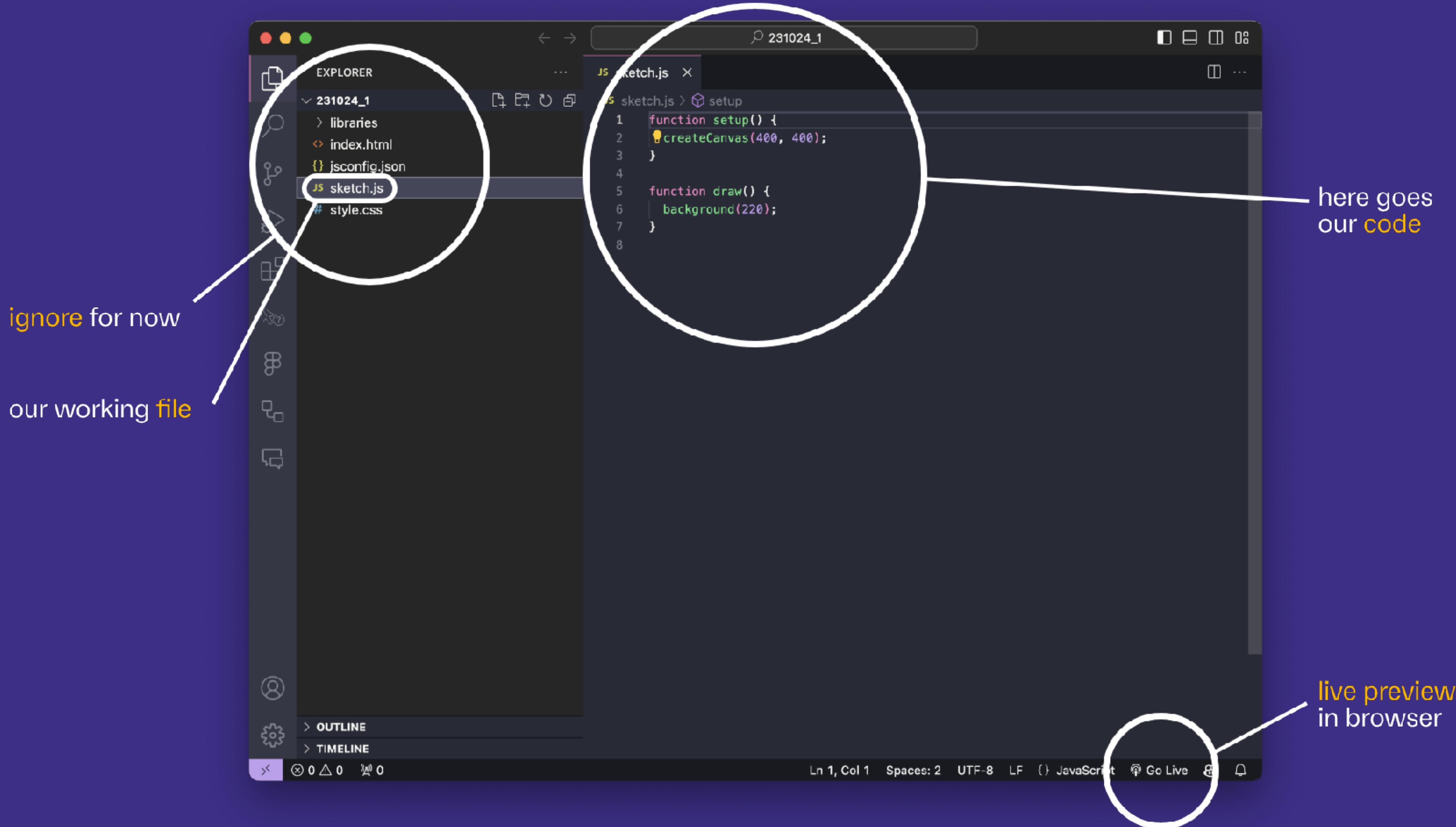
## p5.vscode (Sam Lavigne)



# Create a new p5 project

- cmd + shift + p (mac)  
or view/command palette
- type create p5.js project
- create a new folder  
and hit open



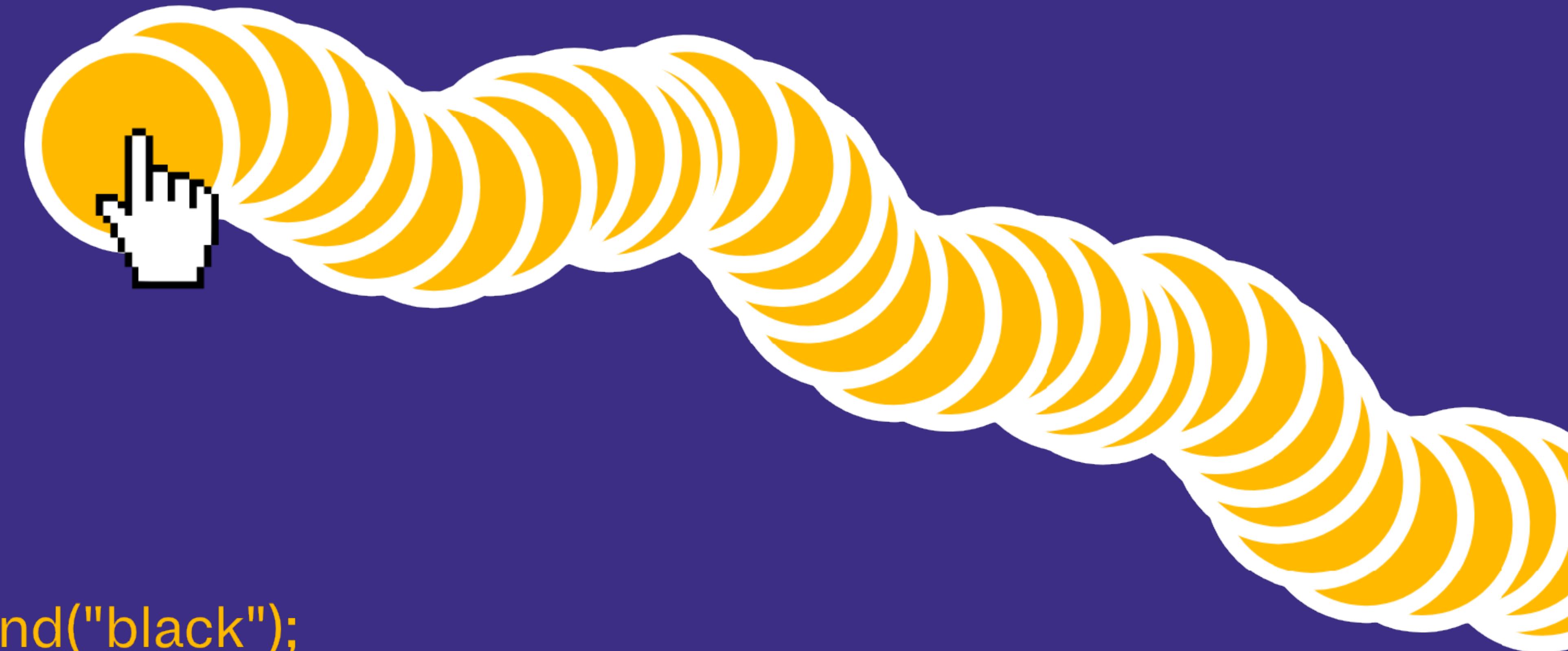


# Our starter template

```
function setup() {  
  createCanvas(400, 400);  
}  
  
function draw() {  
  background(220);  
  
  // this is a comment  
}
```

- **setup()**  
runs only once
- **draw()**  
loops continuously  
(60 times per second)
- **createCanvas()**  
our window size  
(width and height)
- **// comment**  
ignored by the program

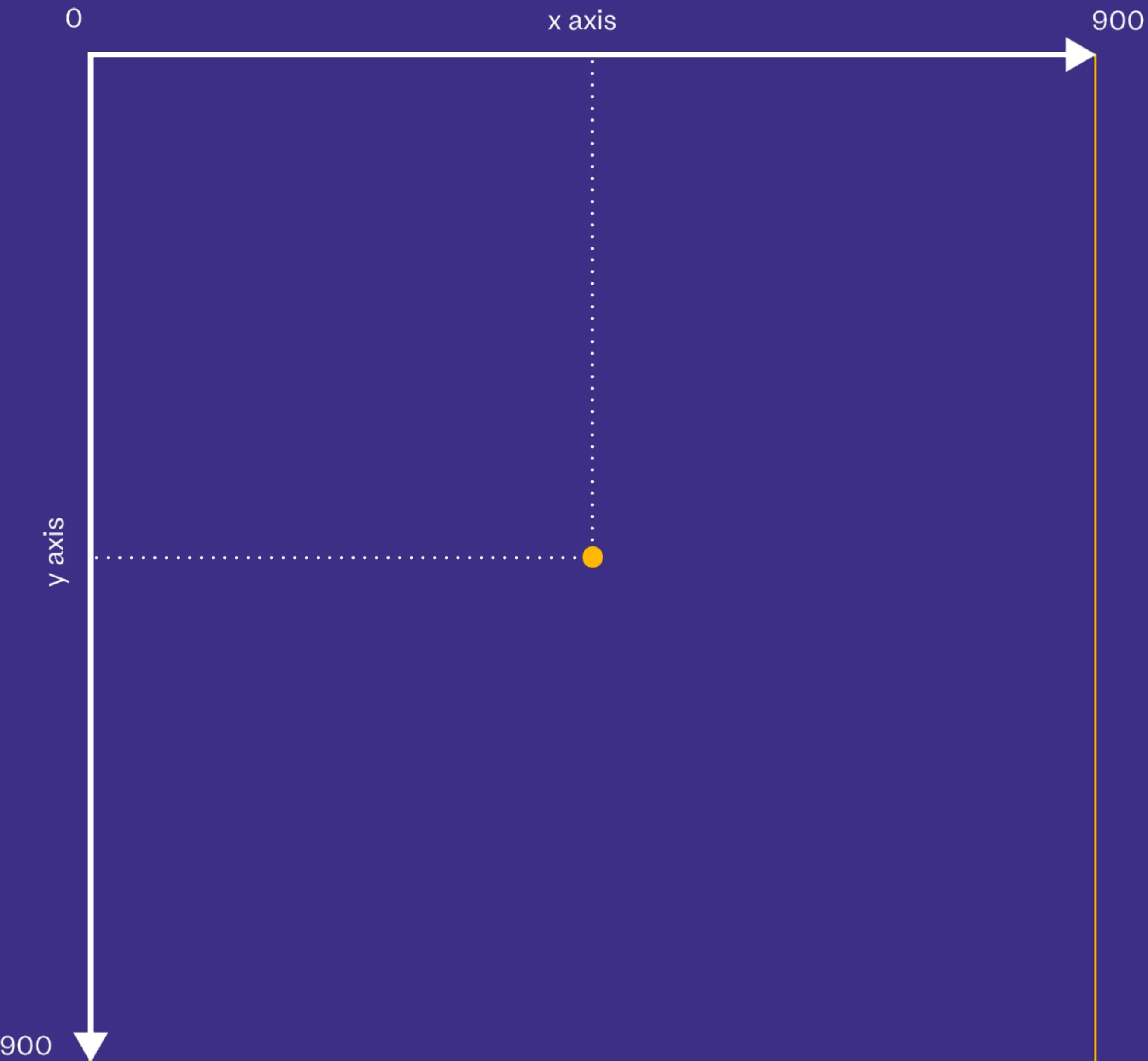
test, test, test



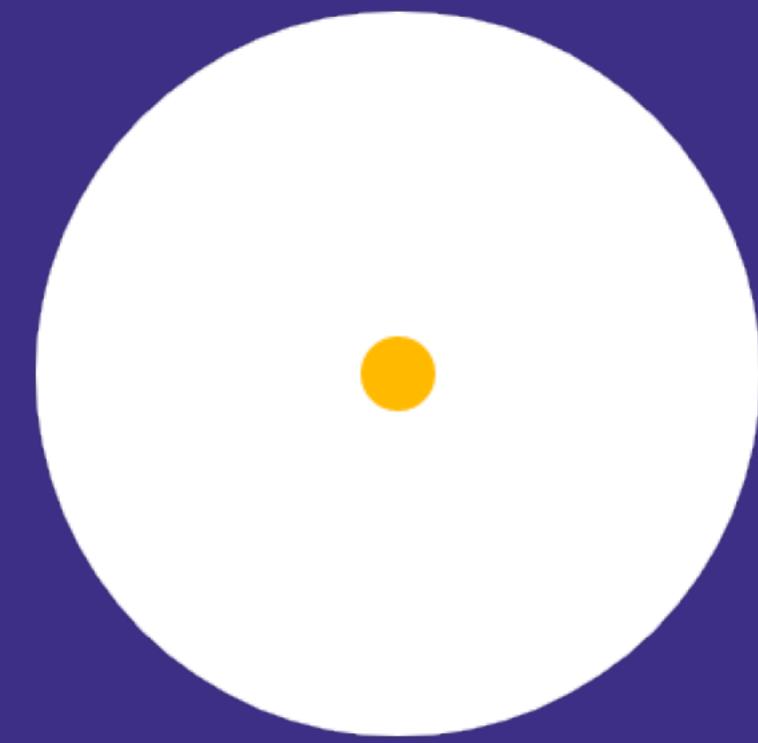
```
// background("black");
ellipse(mouseX, mouseY);
```

# Coordinate system

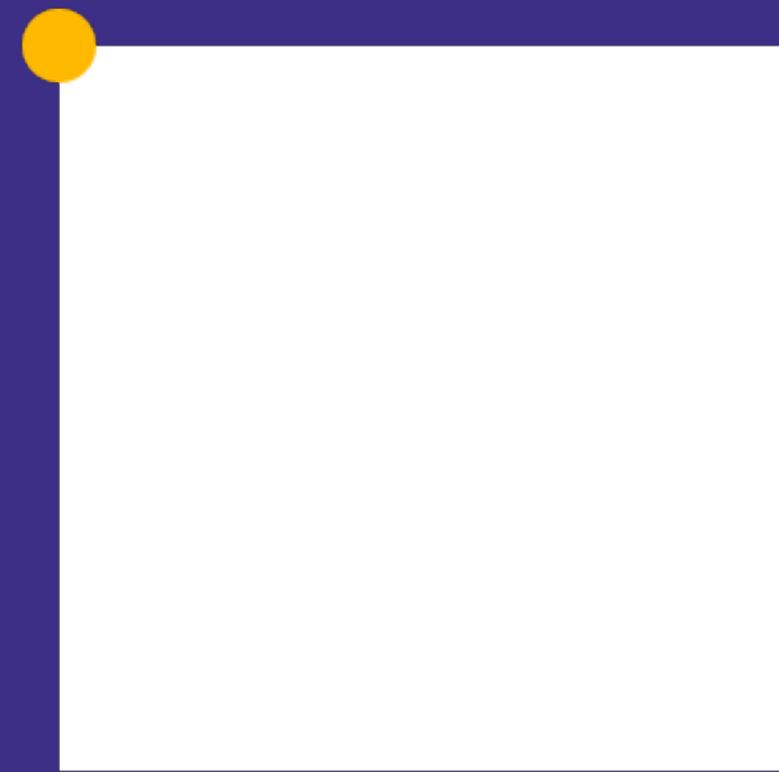
```
point (450, 450);  
point (width/2, height/2);
```



# Basic geometric shapes



`ellipse(x, y, w, h);`

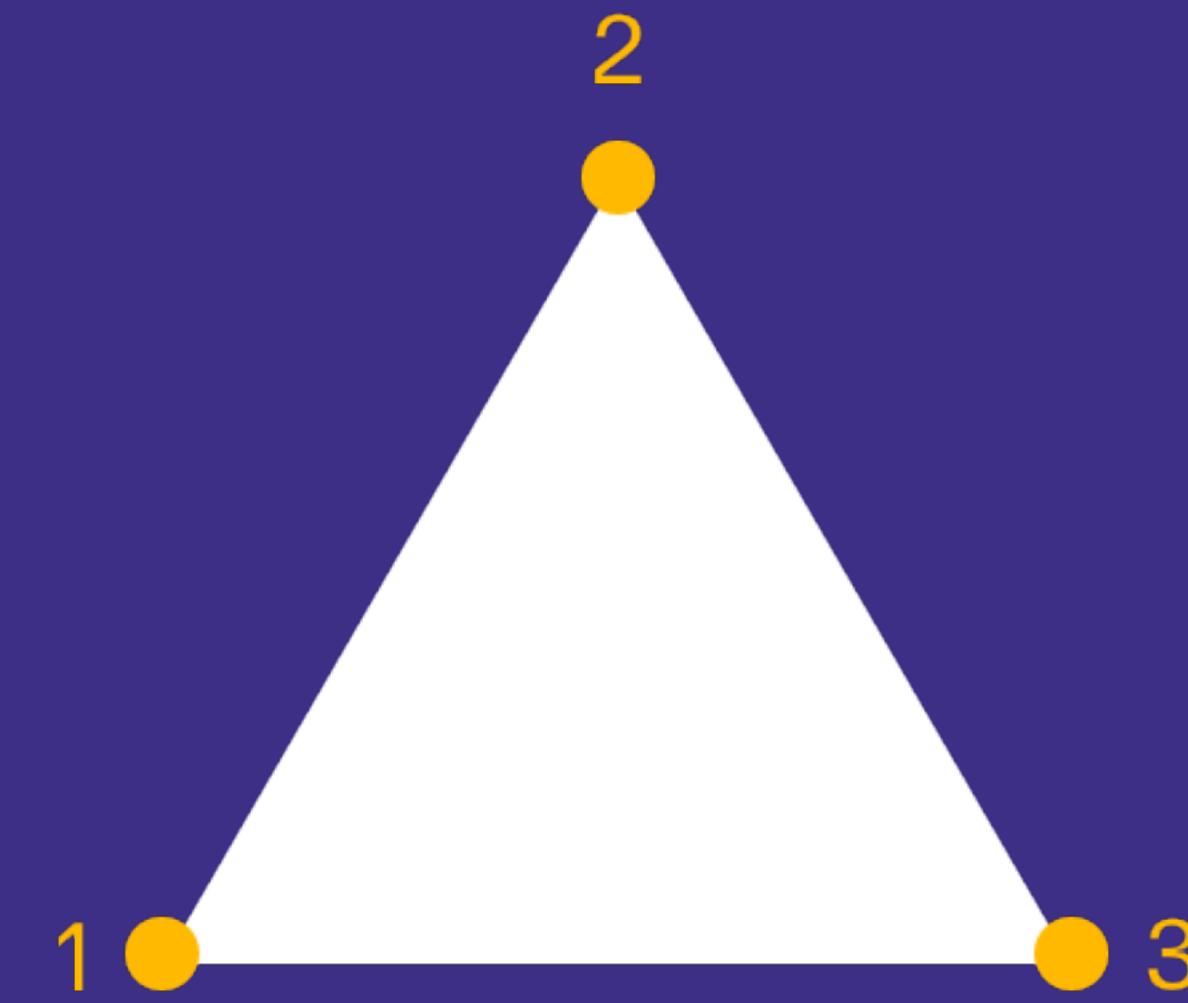


`rect(x, y, w, h);`

# Basic geometric shapes



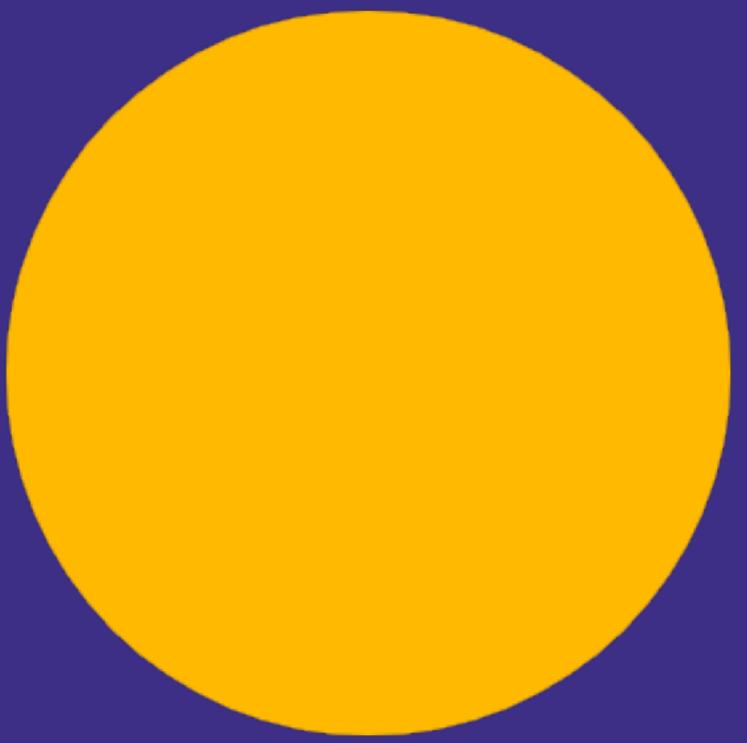
```
line(x1, y1, x2, y2);
```



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

# Colors & styling

hex color picker  
<https://g.co/kgs/9h4Esb>



```
fill("#FFBA00");  
noStroke();
```



```
stroke("#FFBA00");  
strokeWeight(20);  
noFill();
```

# Flow of the code

```
draw blue background;
```

```
    draw a rectangle;
```

```
        take green color;
```

```
            draw a circle;
```

```
        take blue color;
```

```
            draw a triangle;
```

→ last line gets rendered on top (like the highest Photoshop layer)

→ we draw inside the **draw()** loop

→ always close your brackets!  
() [] {}

→ separate functions with semicolons;  
;

# Structure your code with comments

```
// comments can be used to document and structure your code
// you can also use them to outline and sketch your code before you write it

// here starts our setup function, it only runs once!
function setup() {
    createCanvas(400, 400); // this defines the size of the canvas, width and height
}

// here starts our draw function, it runs 60 times per second!
function draw() {
    // this is our background color
    background(220);

    // here will be yellow square

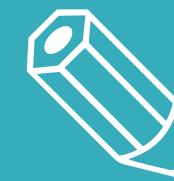
    // here will be a blue triangle
}
```

# Reference

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

The screenshot shows the p5.js Reference website. At the top left is the p5.js logo. To its right is a navigation bar with links: Home (red), Editor, Download, Donate, Get Started, Reference, Libraries, Learn, Teach, Examples, Contribute, Books, Community, and Showcase. Below the navigation bar is a search bar labeled "Search reference". The main content area features a grid of categories: 3D, Color, Constants, DOM, Data, Environment, Events, Foundation, IO, Image, Math, Rendering, Shape, Structure, Transform, and Typography. Under the "Examples" section, there's a table comparing the "Environment" and "Color" libraries. The "Environment" column lists: describe(), describeElement(), textOutput(), gridOutput(), print(), frameCount, deltaTime, and focused. The "Color" column lists: Creating & Reading, alpha(), blue(), brightness(), color(), green(), hue(), Setting, background(), clear(), colorMode(), fill(), noFill(), and noStroke().

	Environment	Color
describe()	Creating & Reading	Setting
describeElement()	alpha()	background()
textOutput()	blue()	clear()
gridOutput()	brightness()	colorMode()
print()	color()	fill()
frameCount	green()	noFill()
deltaTime	hue()	noStroke()
focused		



# Tasks

- 👉 switch background on/off
- 👉 put an emoji on the mouse
- 👉 put an image on the mouse
- 👉 put a rectangle on the mouse
- 👉 document your experiments with screenshots / screenrecord

# Thank you :-)



IG @nahuel.gerth  
GitHub/nahuelGerthVK

info@nahuelgerth.de  
www.nahuelgerth.de