

CRITICAL MAKING FOUNDATIONS

FINDING YOUR SUPERPOWER THROUGH  
DIGITAL AND PHYSICAL MAKING

APRD 5005 | FALL 2021

INSTRUCTOR: RJ DURAN

MAKE.CMCI.STUDIO



# Agenda

## **HOURL 1**

Course website & Slack

CM courses this year

Class structure &  
Expectations

Schedule

Making as a practice

10 min break

## **HOURL 2**

Intro to Processing

Sketching in code

10 min break

## **HOURL 3**

Making making a practice  
through "dailies" or  
"everydays"

Assignment 1

*COURSE*

# Course Website

<https://make.cmci.studio>

The class **website** contains our schedule, syllabus, course resources, assignments, and projects.

Slack Channel: **#cmstudio**

**slack** will be our primary communication channel



# Course Themes

Building creative habits and practices

Deconstructing problems and exploring solution  
spaces

Analyzing systems of logic

Development and design skills building

Rapid prototyping toolsets

Digital fabrication workflows

Emerging tech

## IMMERSIVE MEDIA DESIGN

### DESIGNING FOR EXTENDED REALITIES

APRD 5007 | SPRING 2022  
INSTRUCTOR: RJ DURAN  
MAKE.CMCI.STUDIO



Tools for XR = VR, AR, MR...R?

Media for XR

Digital Asset Workflows

UX of XR

Scripting / Programming

Interactivity

## HARDWARE UX

### DESIGNING CONNECTED, RESPONSIVE, AND INTERACTIVE EXPERIENCES

APRD 5019 | SPRING 2022  
INSTRUCTOR: RJ DURAN  
MAKE.CMCI.STUDIO



Interactivity through hardware based systems

Human computer interaction

Low cost micro-controllers and single board computers

Input and output devices

Electronics prototyping

# What is this class about?

Developing awareness *of* and control *over* your creative process.

Learning how tools and technology enable you to make what you envision.

Making things. Every week.

Exploring different perspectives in relation to technology.

Becoming a better collaborator and team member.

Trying stuff that might (scare || intimidate || confuse) you.

Making it what you want it to be.

# Expectations

Asking questions and sharing work every week.

Making things that help you answer questions and ask new questions.

*(Making things is better than not making things.)*

Doing research on related topics that interest you.

Sketching out ideas and concepts to build.

Engaging with each other via Slack and work groups. Dedicate time to work together.

Having fun! Being flexible and adaptable. Learning to be malleable in your making.

*SCHEDULE*

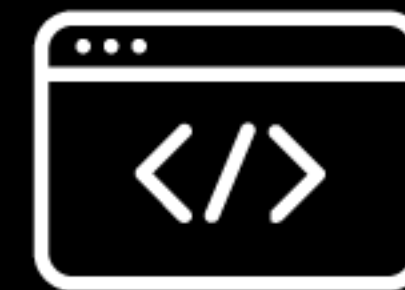
# Digital

10100  
00101  
10100

*code*



*script*



*app*

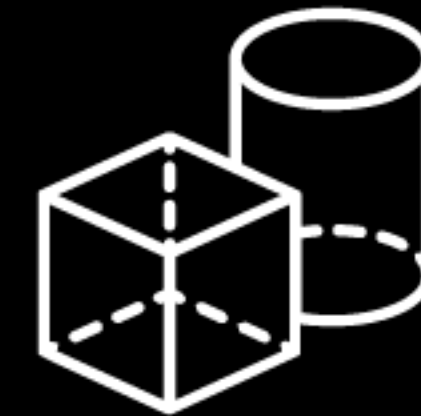
# Physical



*sketch*



*cad*

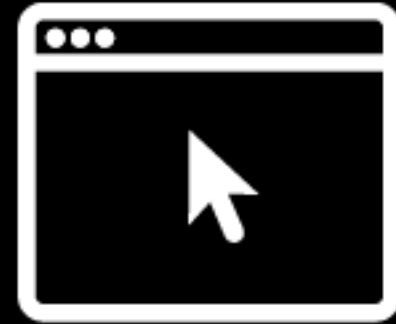


*form*

# Final Project



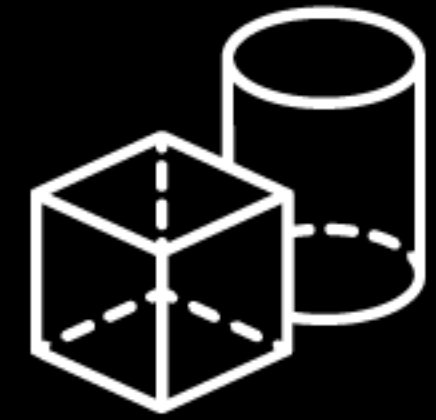
*experience*



*web app*



*mobile app*



*physical*



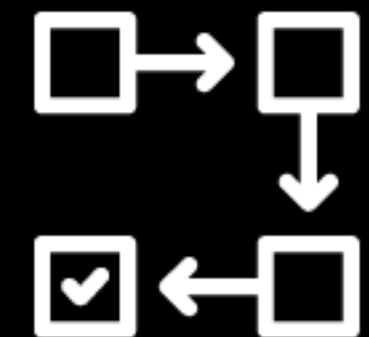
*software*



*product*



*artwork*



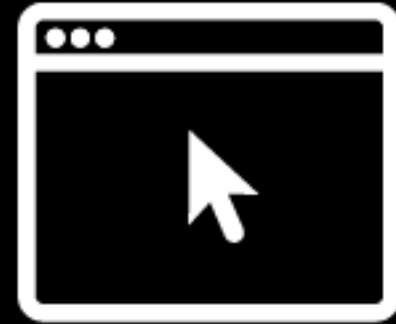
*process*



# Final Project



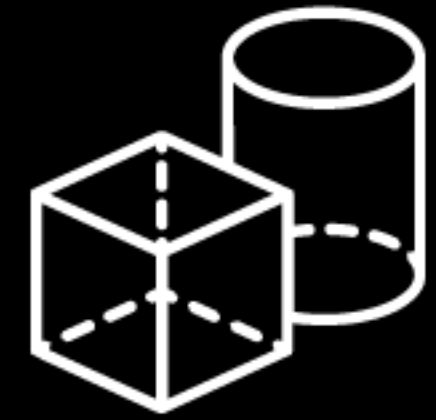
*experience*



*web app*



*mobile app*



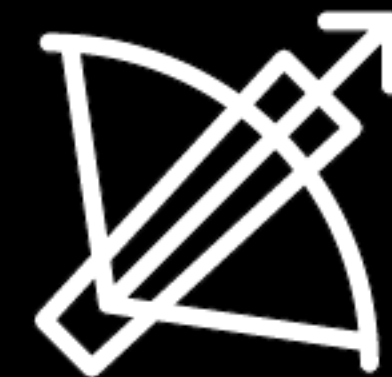
*physical*



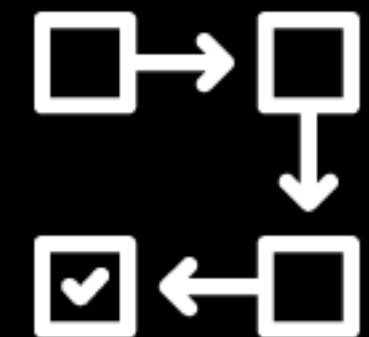
*software*



*product*



*marshmallow  
crossbow*



*process*

## Maker Blogs

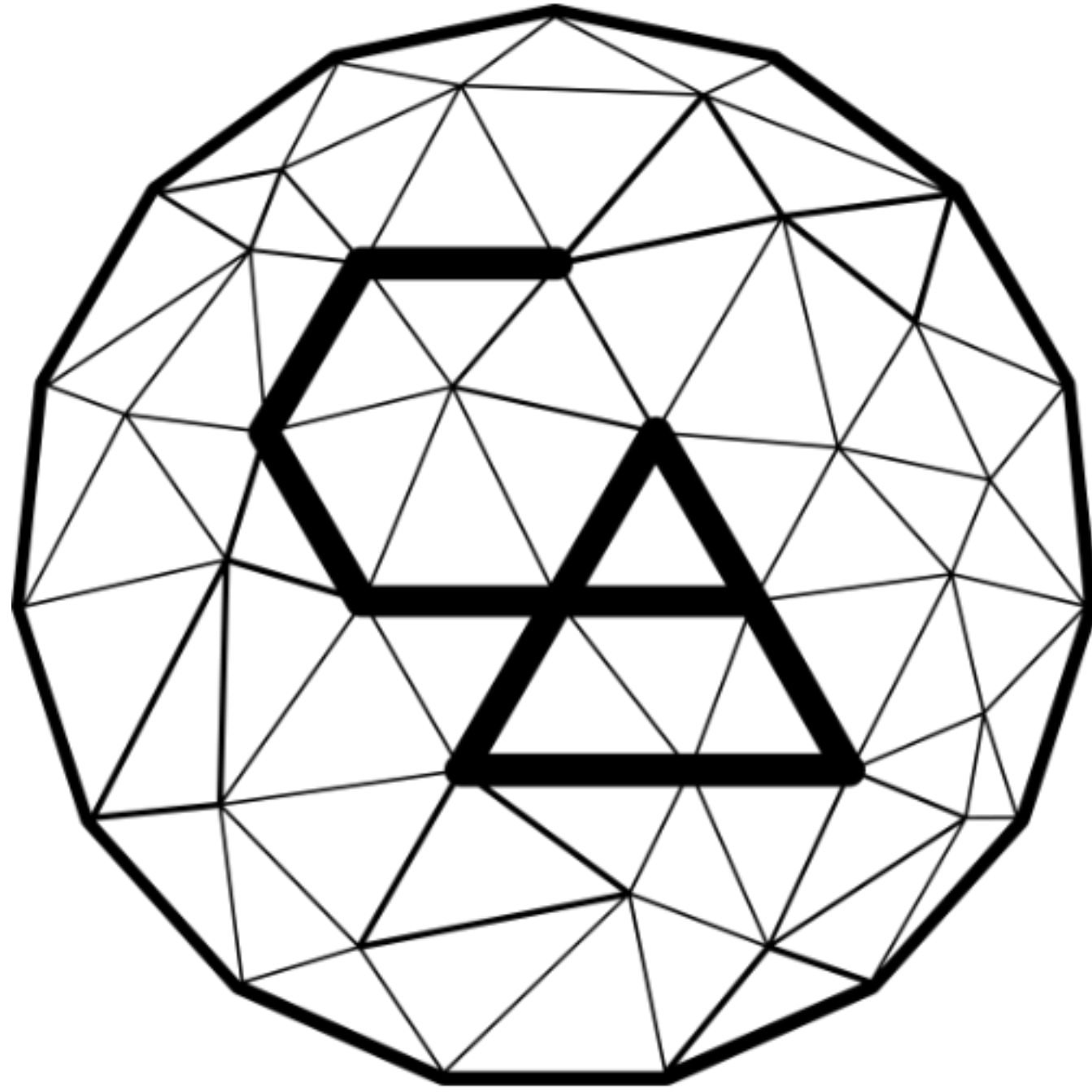
*make.cmci.studio/students*

## Projects

*make.cmci.studio/projects*

## Class Archive

*make.cmci.studio/archive*



**CREATIVE  
APPLICATIONS  
NETWORK**

# Deliverables

## Assignments

*submitted weekly,  
reviewed / discussed in class*

## Final Project

*proposal due by Friday, Nov 12  
presentations on Monday, Dec 6*

*MAKING AS A PRACTICE*

*Make if you know what to make.*

*Make if you don't know what to make.*

*If you want to make stuff  
you have to get comfortable with being uncomfortable*

*~ David Slayden (2010...maybe...probably)*

*Learn to improvise solutions on the fly.*

*How?*

*Use whatever works for now.*

*@profg.co*    *(maybe MacGyver too)*



*Creativity is a process, not a state of mind.*

*~ Manohar Vanga*

*Nothing is a mistake.*

*There's no win and no fail, there's only make.*

*~ John Cage*

<https://bit.ly/1FS5XBO>

*What did you find challenging?*

*Making – I'm good at coming up with an idea, but find myself surprised every time at the amount of work it takes to realize that idea. I think my head exploded about a dozen times when trying to navigate through a new digital tool (rhino 3D, javascript, etc.). I would find myself easily frustrated when the tech wouldn't work as I wanted. It required a ton more research and learning to finally realize my idea.*

*~ Mark Grundberg (C11)*

<https://medium.com/re-write/final-re-post-3a32f4da7673>

*Ask questions, lots of them, when in critical making, you are not expected to know everything.*

*Don't take critique personally, receive it with gratitude.*

*And find ways to connect with and lean on your classmates. Especially when isolated at home.*

*~ Mark Grundberg (C11)*

*You are here to learn, so be prepare. There are projects in a different pace environment. For entrepreneurial design, you will have a year to accomplish, and for Critical making, you might only have a week to make. Don't feel stressed out; ask your instructor for help. They will give you great advice on how to solve it.*

*Time management is important. I've worked 20+ hours and studied full-time for the whole year. You just have to know when to turn on your switch. And please do you work ahead of time. Don't wait for the last minute, especially Critical Making. It will not work the way you think. Be prepared for five back up plans.*

*Communicate with your classmates. Use Slack, Figma, Miro, Trello, Google Suites and Zoom. There are so many great tools to communicate and collaborate. Hint hint, look for student discounts.*

*Organize and document your work. I highly recommend you to take many visuals as possible, if you don't want to recreate them all. Take notes, sketches, pictures, screenshots, and videos.*

*~ Yeon Choi (C11)*

*My advice for next year's cohort going into CM1: get excited to be uncomfortable. What I actually mean by this is don't diminish or write off areas that are not in your wheelhouse or don't seem like a path you're interested in. Those should actually be the weeks you're most excited for because it's an opportunity to learn. It sounds a bit cliché, but it's true.*

*I came into CM knowing some code and figured I would excel in the digital area. In reality, it was something that was extremely frustrating for me. Instead, I found myself getting more and more excited about the physical projects. So much so that I shifted my final project to one that deals with more physical making.*

*~ Vanessa Frye (C11)*

*BREAK*

# *Intro to Processing*



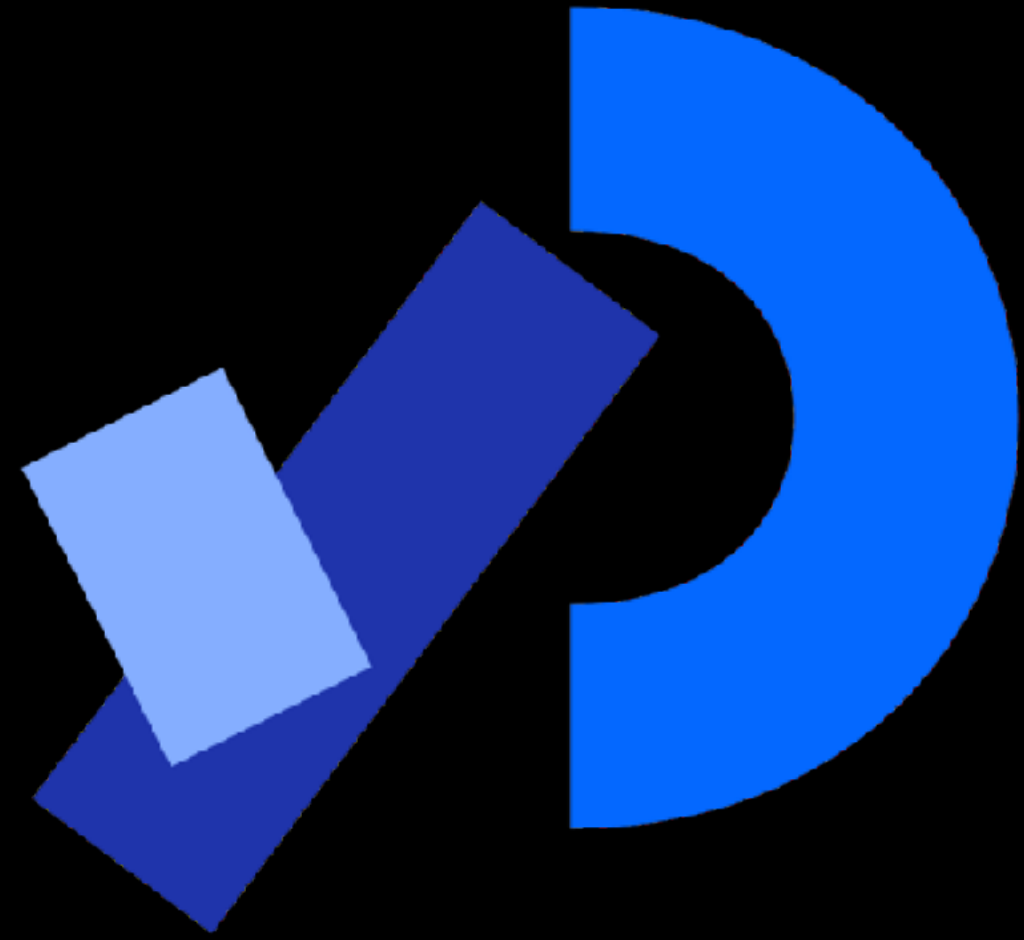
# What is Processing?

Processing is a flexible software sketchbook and a language for learning how to code within the context of the visual arts. Since 2001, Processing has promoted software literacy within the visual arts and visual literacy within technology. There are tens of thousands of students, artists, designers, researchers, and hobbyists who use Processing for learning and prototyping.

*I don't believe that everybody needs to be a programmer or everybody needs to be an engineer, but I think people should have enough of an understanding of how software operates that they can get outside of the constraints imposed by software written for them.*

*~ Casey Reas  
(Co-Creator of Processing)*

*Casey Reas on Coding: You need to be Able to Read and Write*



Java

User Friendly IDE

OS Based

Large Community

Est. 2001

<https://processing.org/>

p5.js

Javascript

Web Editor IDE

Server Based

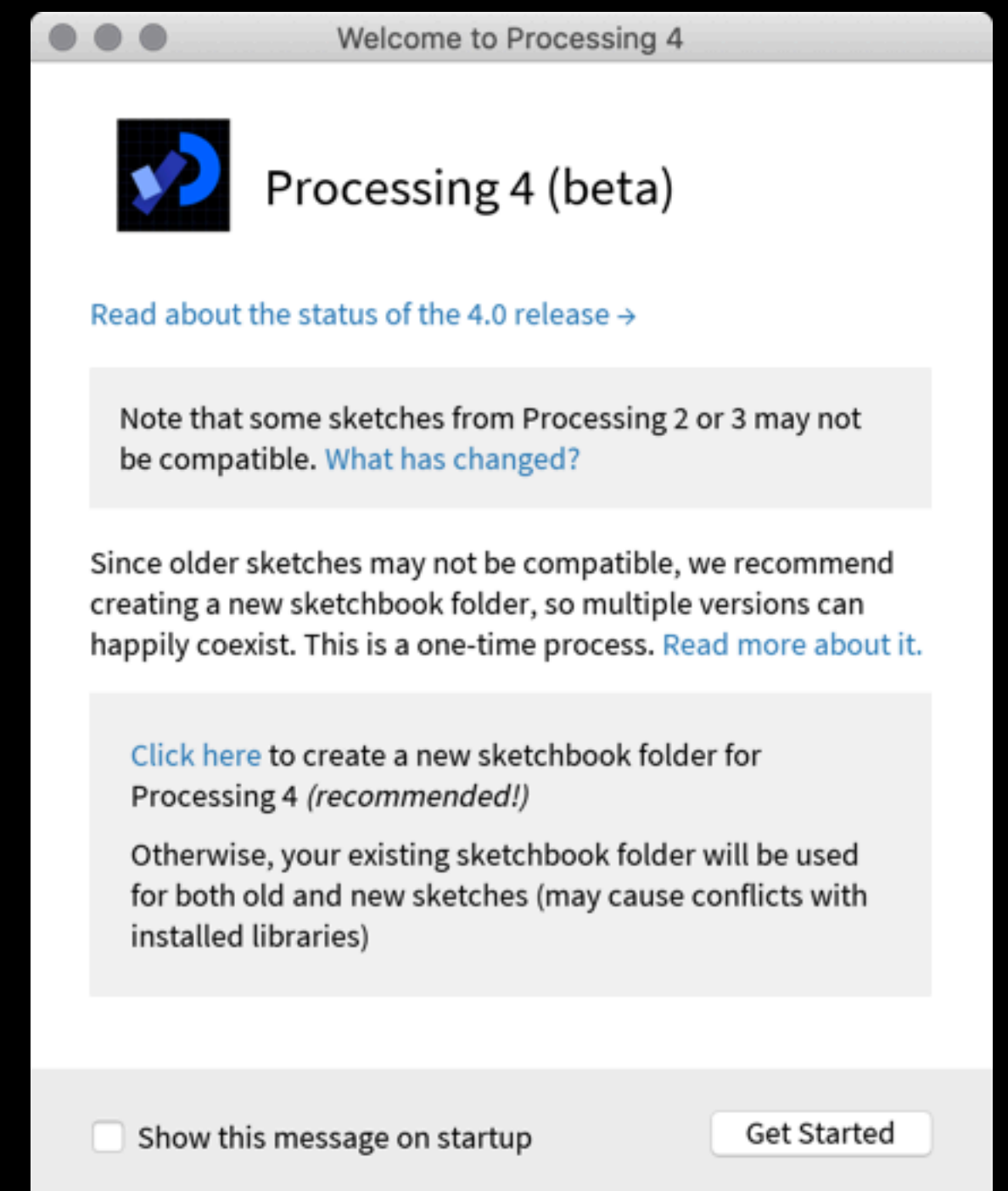
Growing Community

Est. 2014

<https://p5js.org/>

# Install IDE

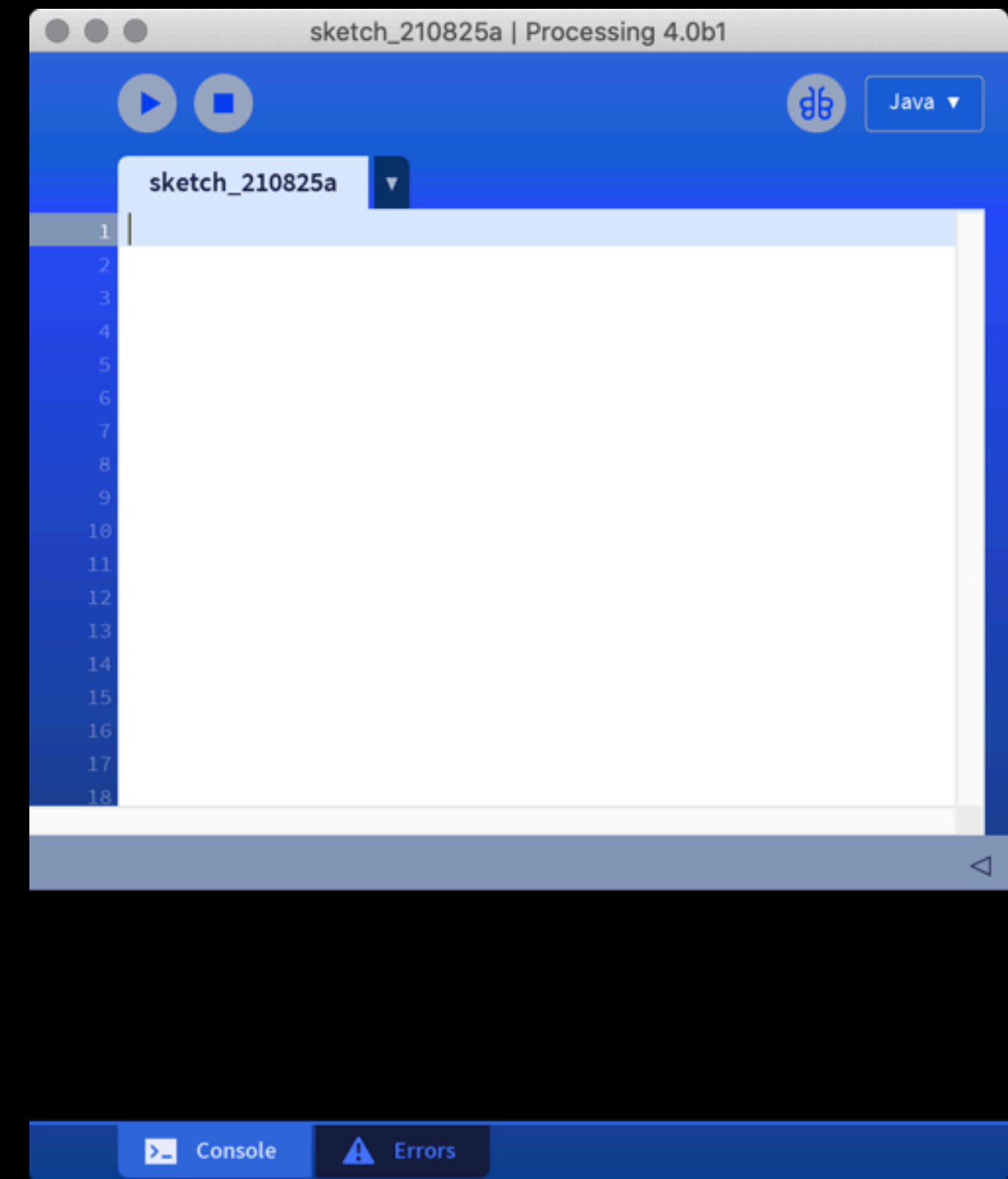
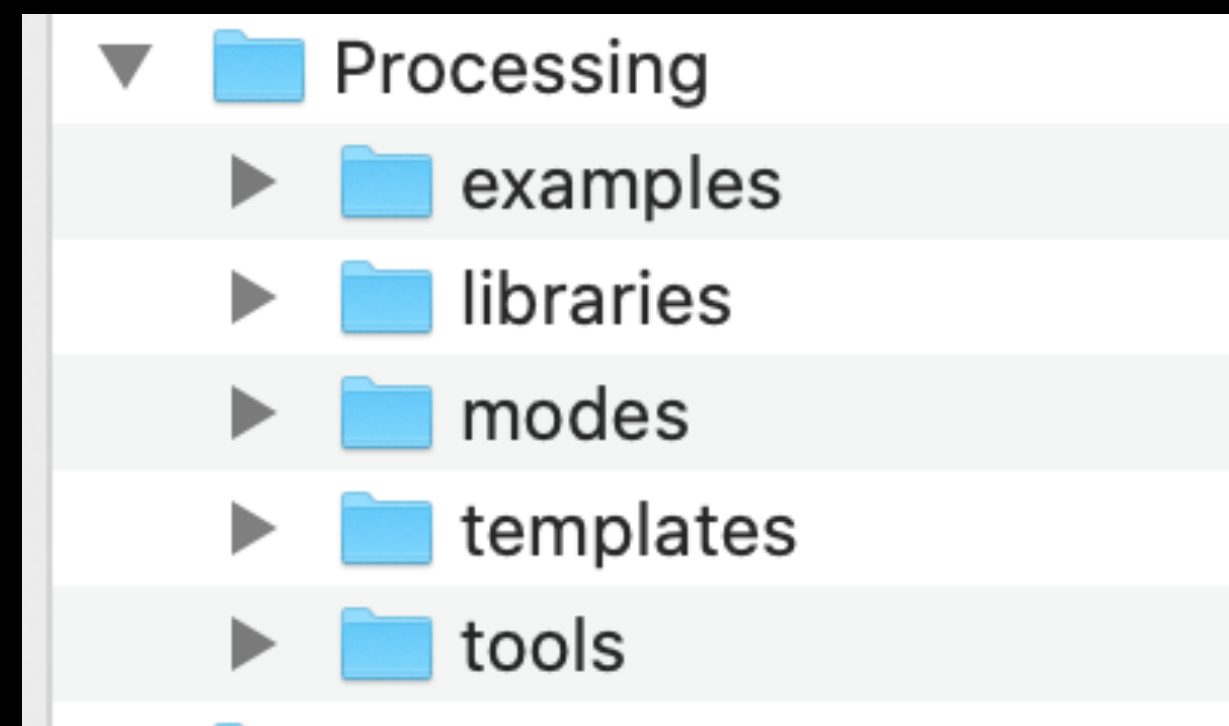
1. Goto <https://processing.org/> and download Processing IDE zip.
2. Extract the zip and move the **Processing.app** file to your **Applications** directory.
3. Launch Processing.app. You will see a popup like this -> It's temporary for now.
4. Click where it says "Click Here" and create a new folder called "Processing" inside your /User/username folder. Click Get Started to proceed.



# Install IDE

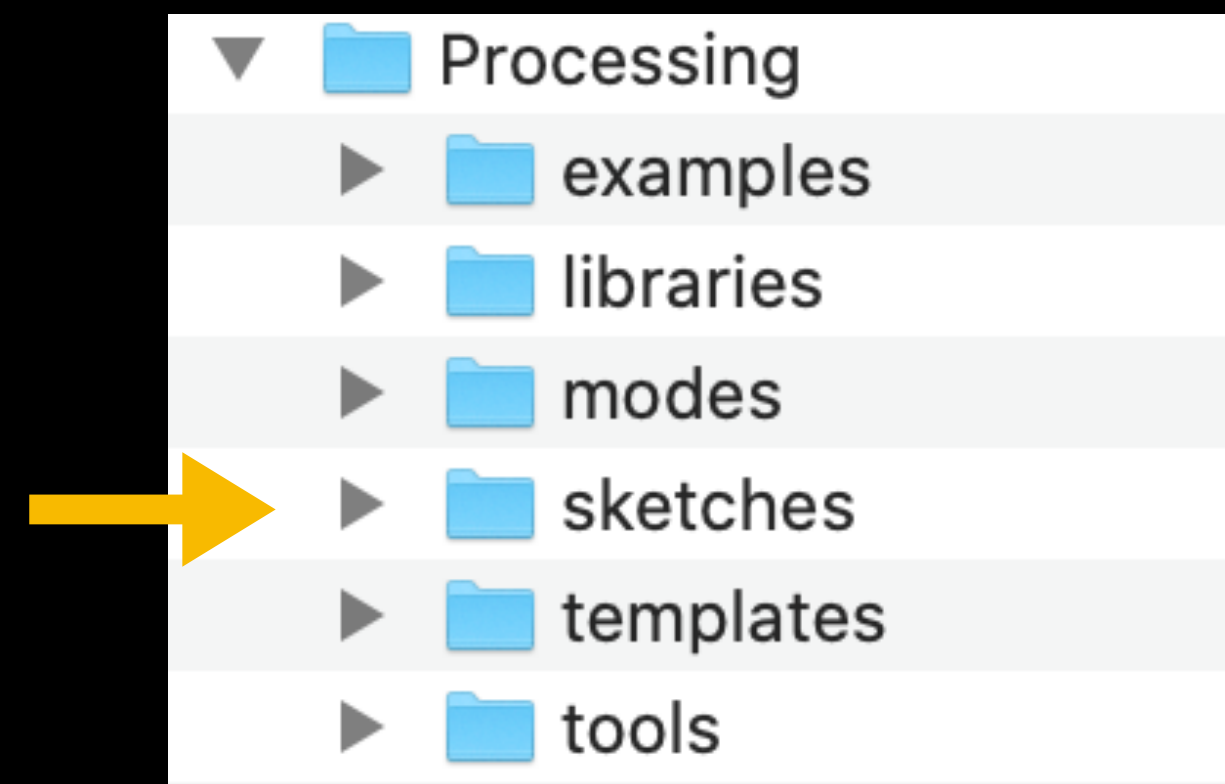
5. You will then see the IDE and can type code into it.

6. Quit and restart Processing to proceed. Restarting the app sets up the “sketch” folder with additional folders used by Processing. Let’s look at that.



# File Management

When it comes time to save a sketch, be sure to create a safe place for your sketches to live. The simplest method is to make a folder called “sketches” inside of your Processing folder.





*How to actually learn any new programming concept*



*Essential*

Changing Stuff and  
Seeing What Happens

# First Sketch

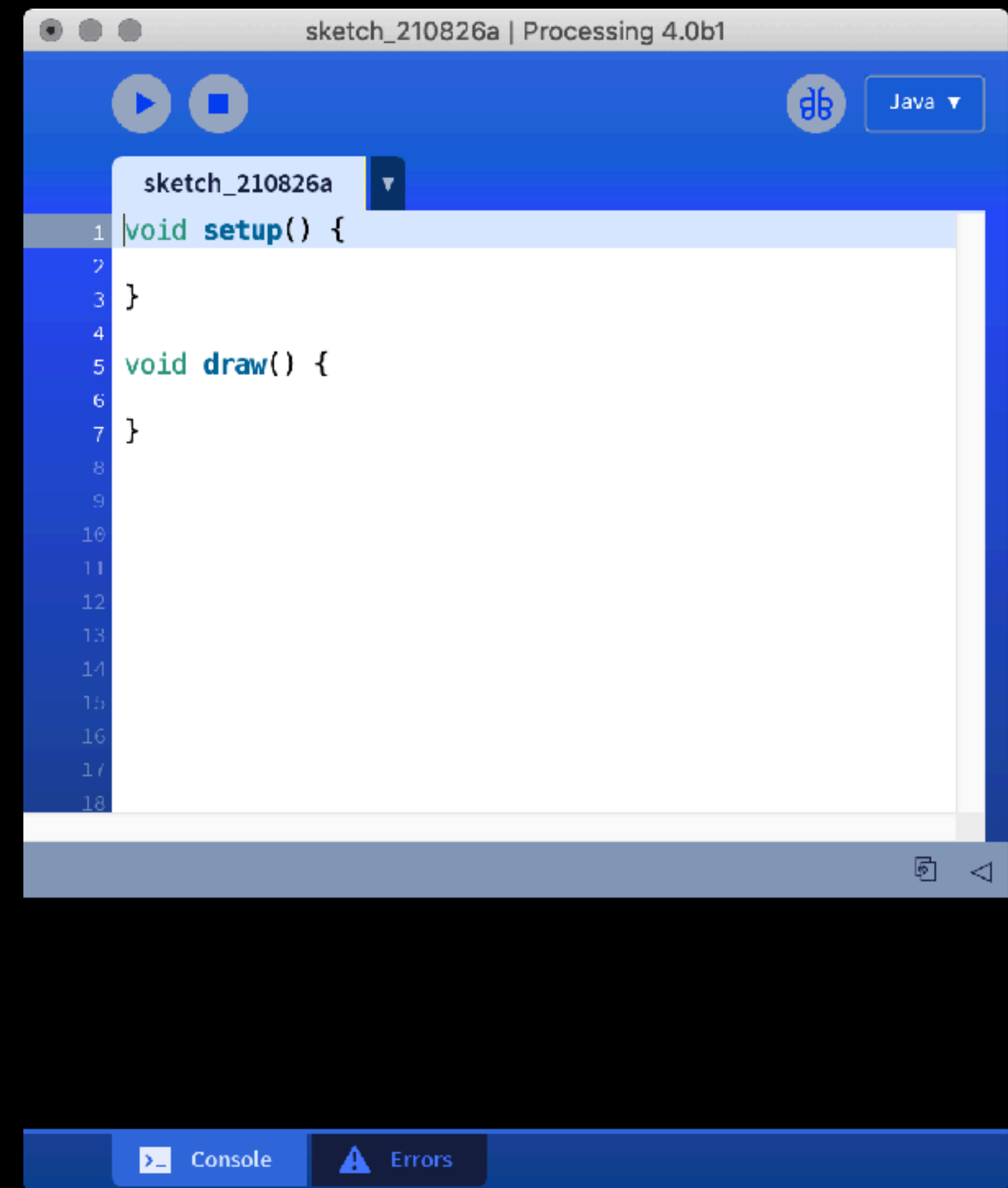
```
void setup() {  
  
}
```

*setup runs one time*

```
void draw() {  
  
}
```

*draw runs until the  
program stops  
(press stop or quit)*

*runs one time per  
frame*





# Challenge #1

On a piece of paper, sketch out a square, circle, and two lines anywhere in one of the boxes on the page. Everything inside of the box is our “sketch”.

After doing this, launch Processing and create a New Sketch.

You are going to translate what you drew by hand into code using Processing.

Feel free to work together.

Tip: You will need to reference the [Processing documentation](#) to find out what functions to use.

# Questions

Where is the sketch origin?

What size is your sketch? What's the width and height in pixels?

How large are your shapes? What are their dimensions?

Where are your objects located within the sketch?

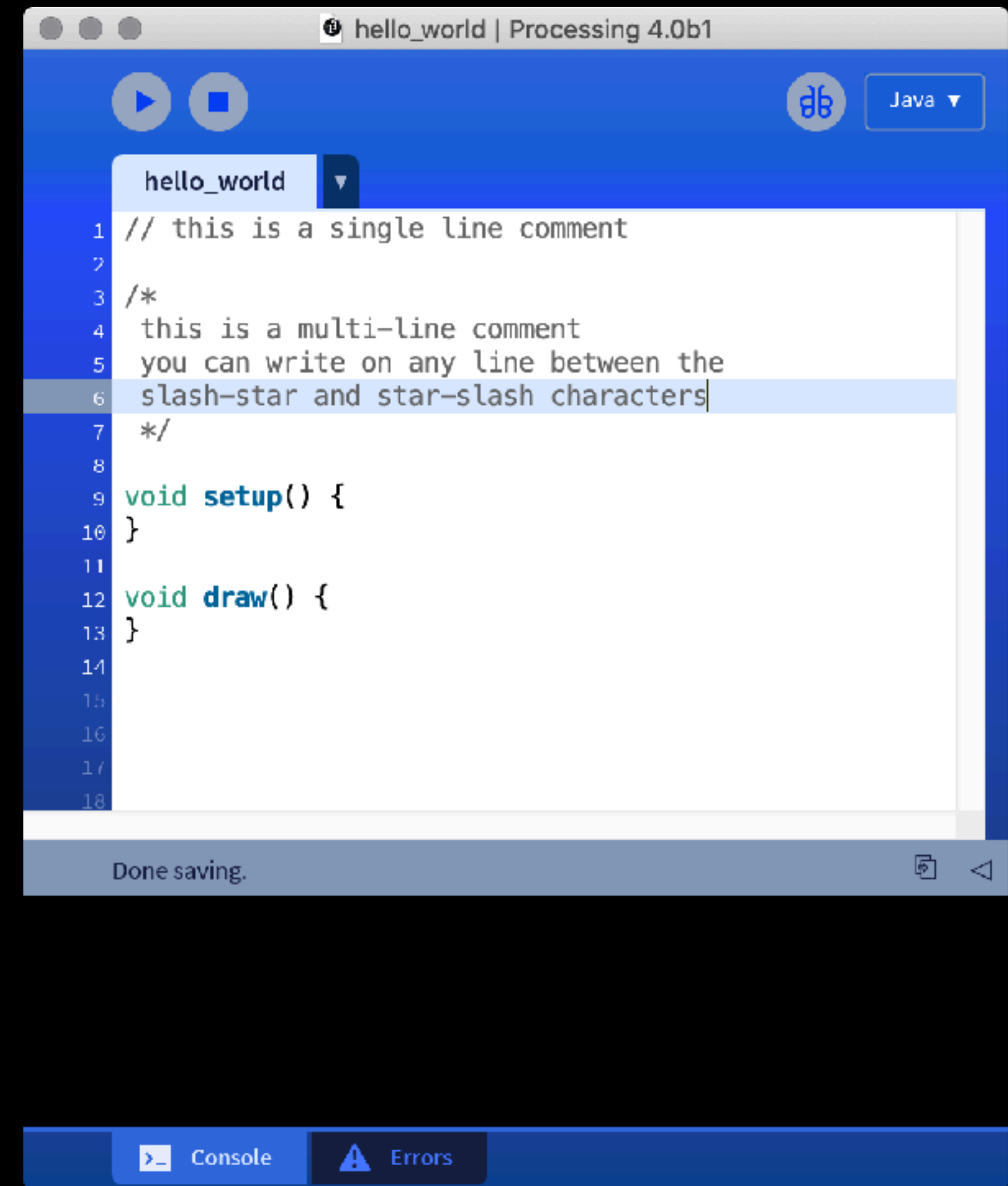
How true to your hand drawn sketch did you make your program?

# Comments

*what are comments used for?*

// single line comment

/\*  
multi-line  
comment  
\*/

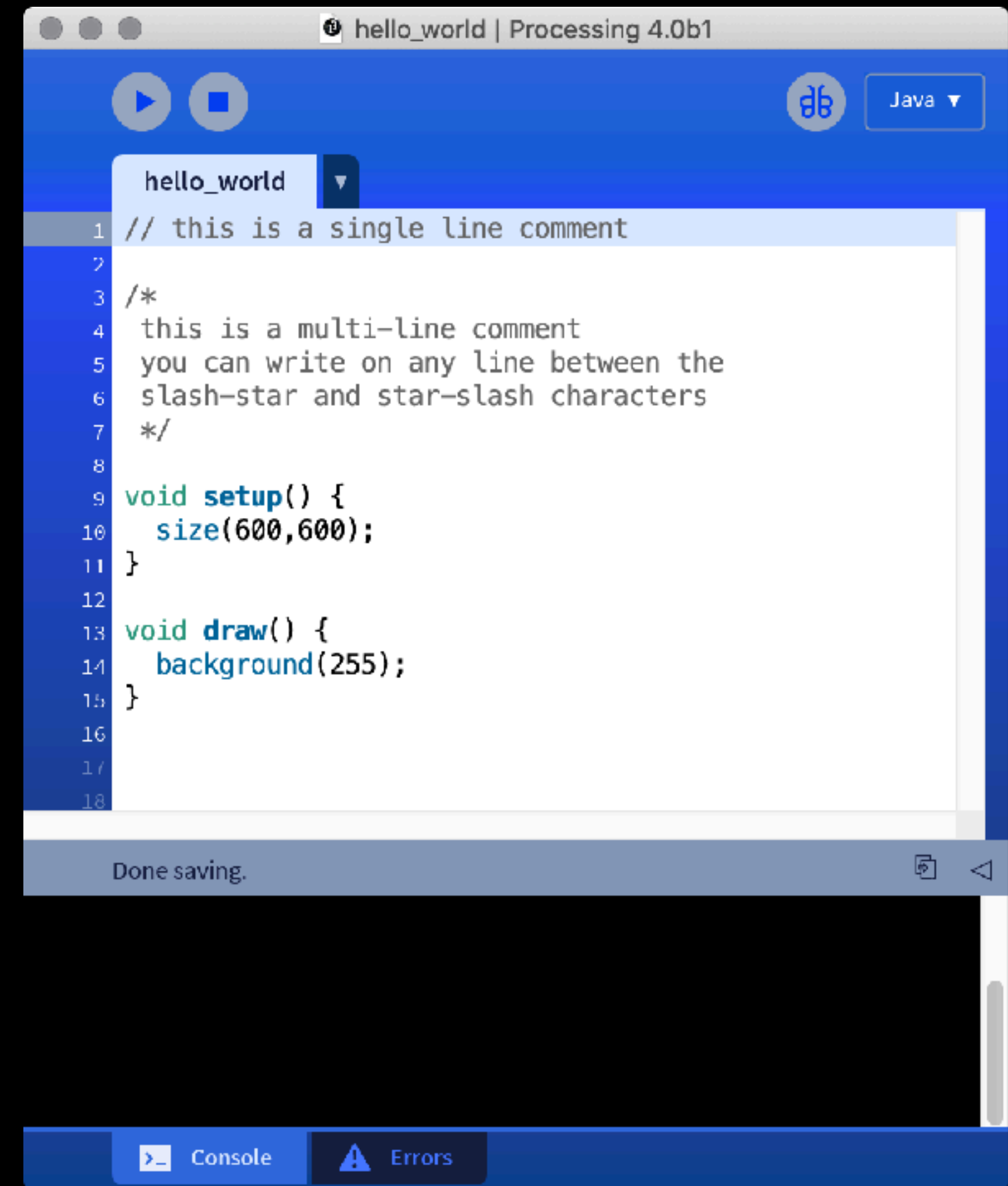


# First Sketch

*what does this code do?*

`size(600,600);`

`background(255);`



# First Sketch

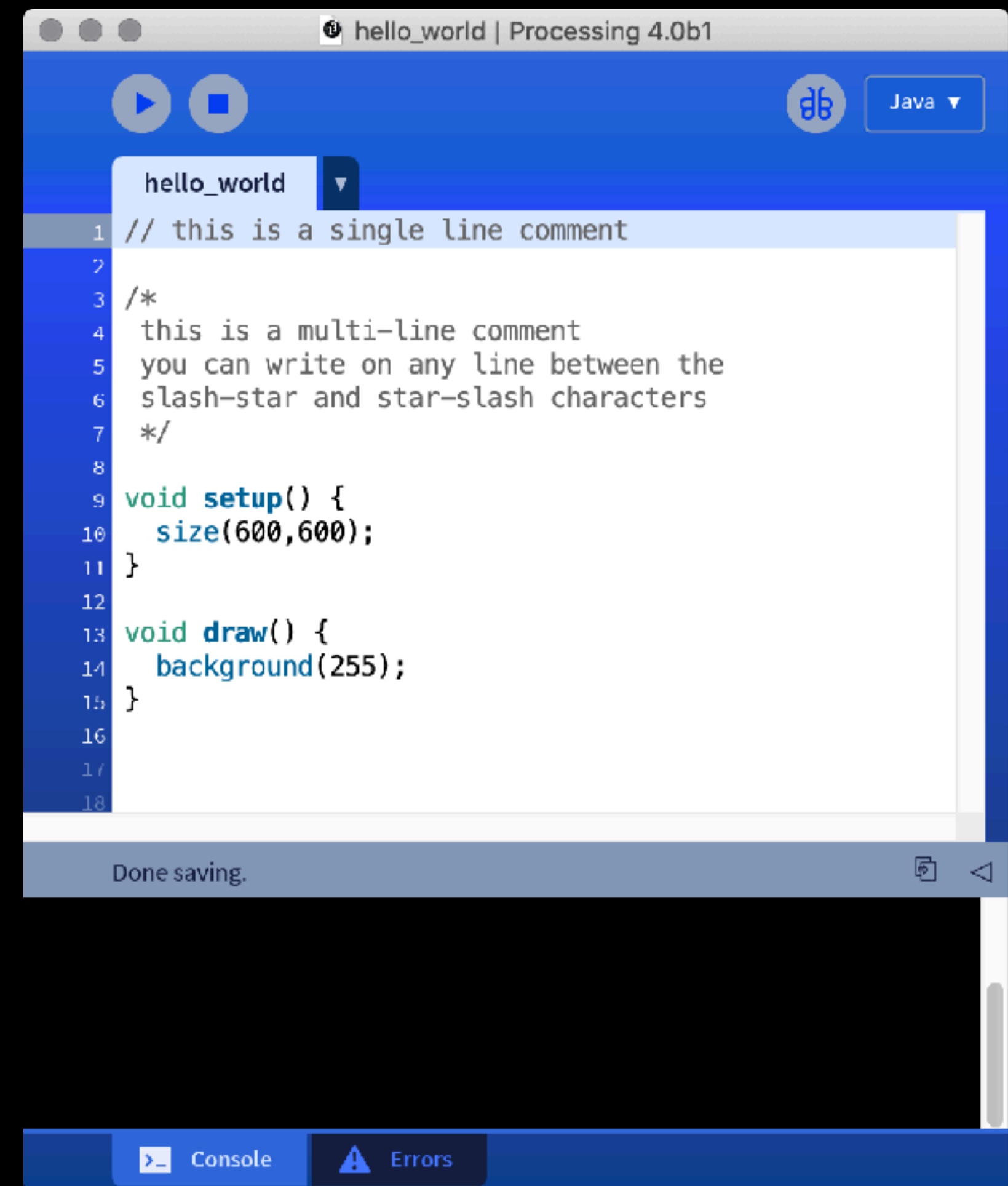
*what does this code do?*

**size(600,600);**

*It sets the size of the sketch to 600 x 600 px*

**background(255);**

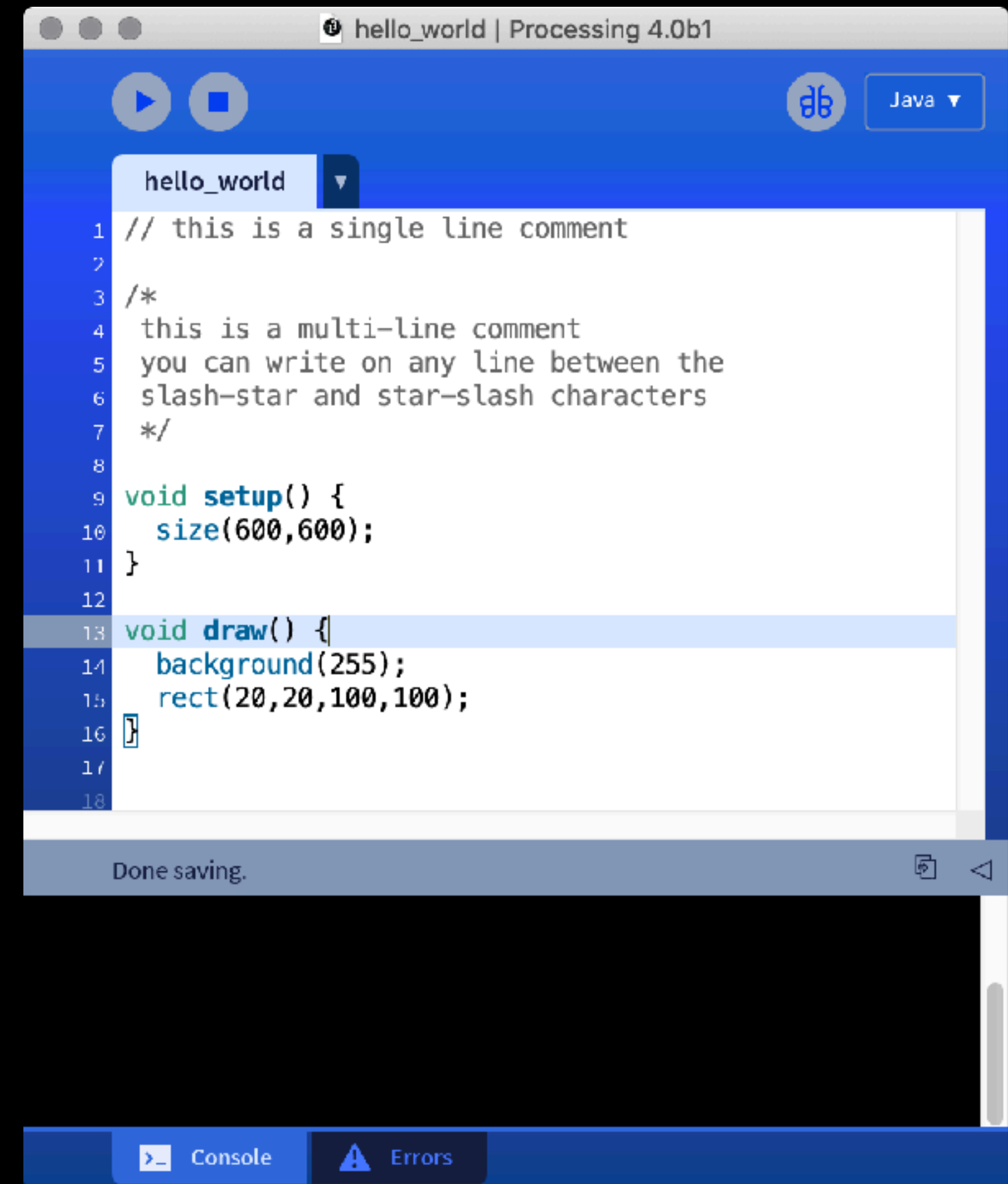
*It sets the background color white.*



# Shapes

*what does this code do?*

```
rect(20,20,100,100);
```

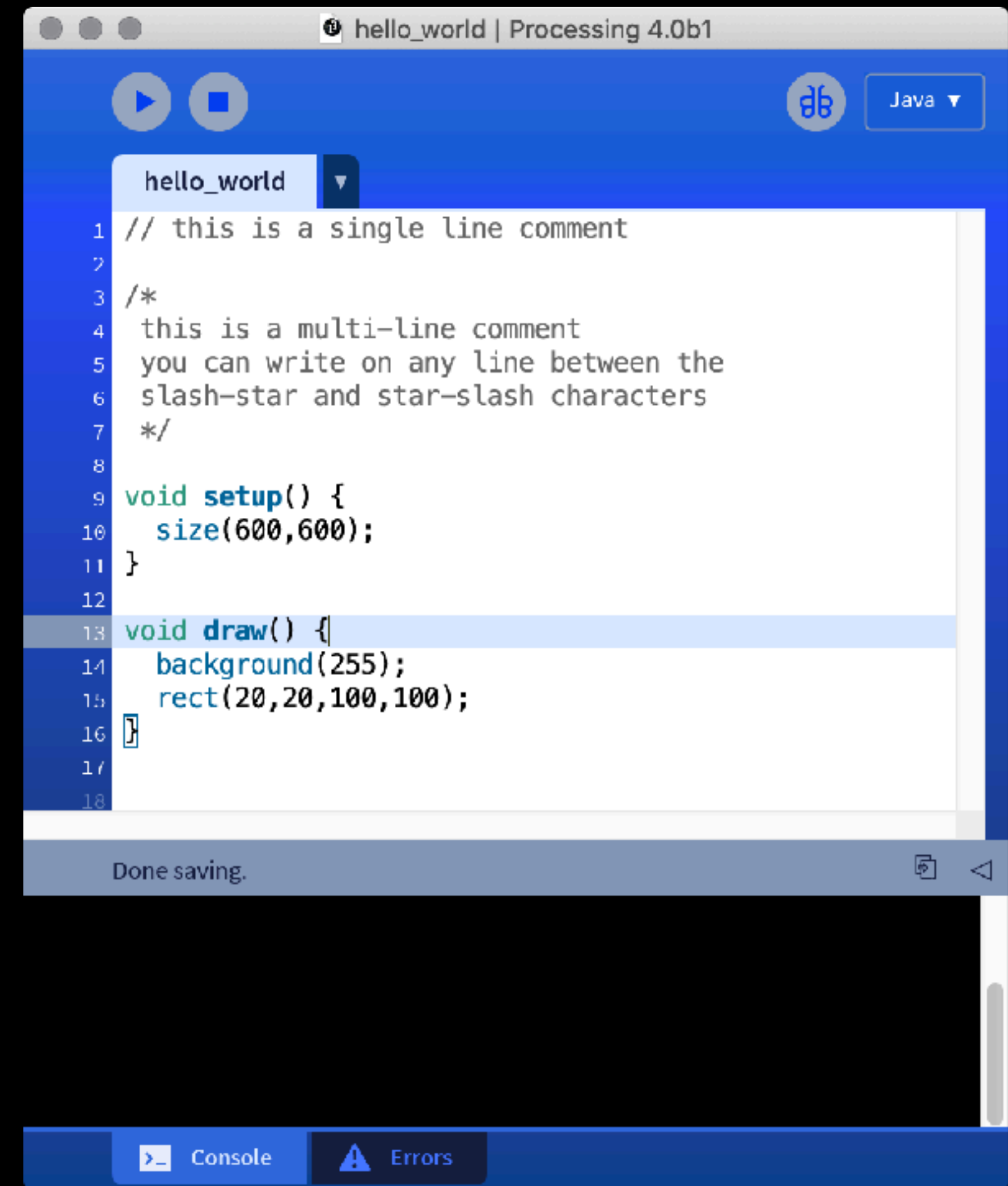


# Shapes

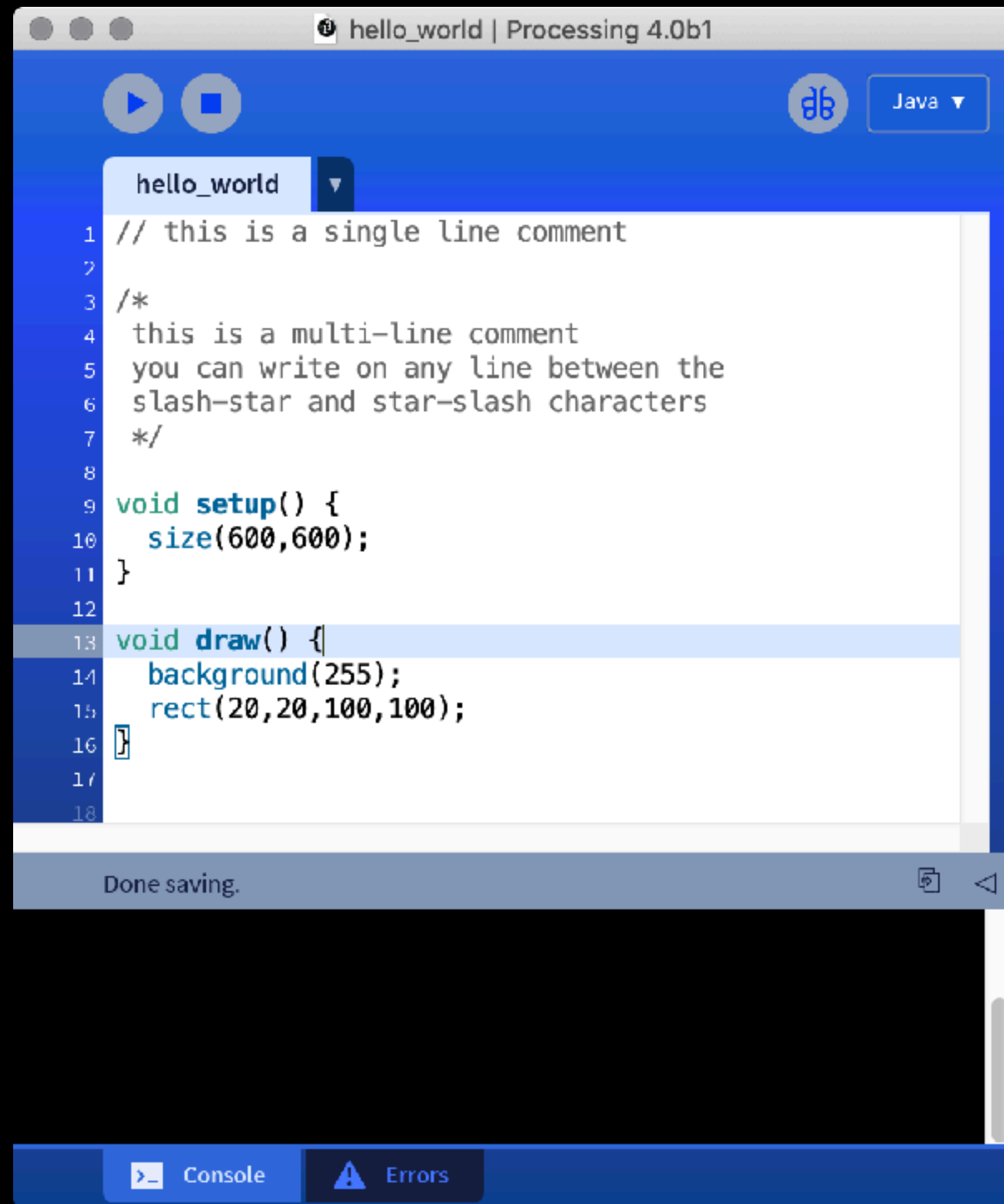
*what does this code do?*

```
rect(20,20,100,100);
```

*It draws a rectangle at  $x = 20$ ,  $y = 20$ , and has a width and height of 100 px. It's a square!*

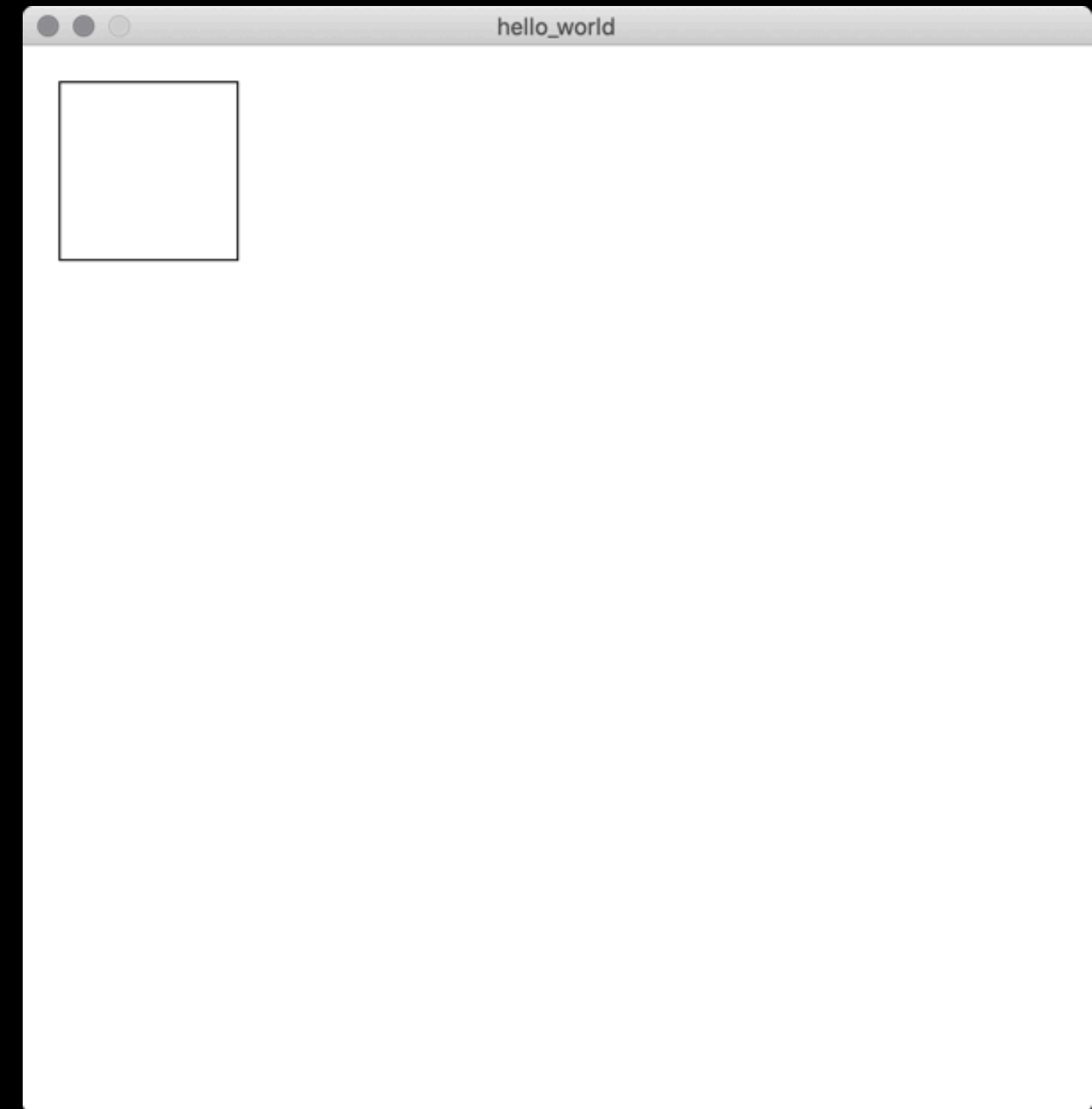


# Output



```
1 // this is a single line comment
2
3 /*
4  this is a multi-line comment
5  you can write on any line between the
6  slash-star and star-slash characters
7  */
8
9 void setup() {
10   size(600,600);
11 }
12
13 void draw() {
14   background(255);
15   rect(20,20,100,100);
16 }
17
18
```

Done saving.



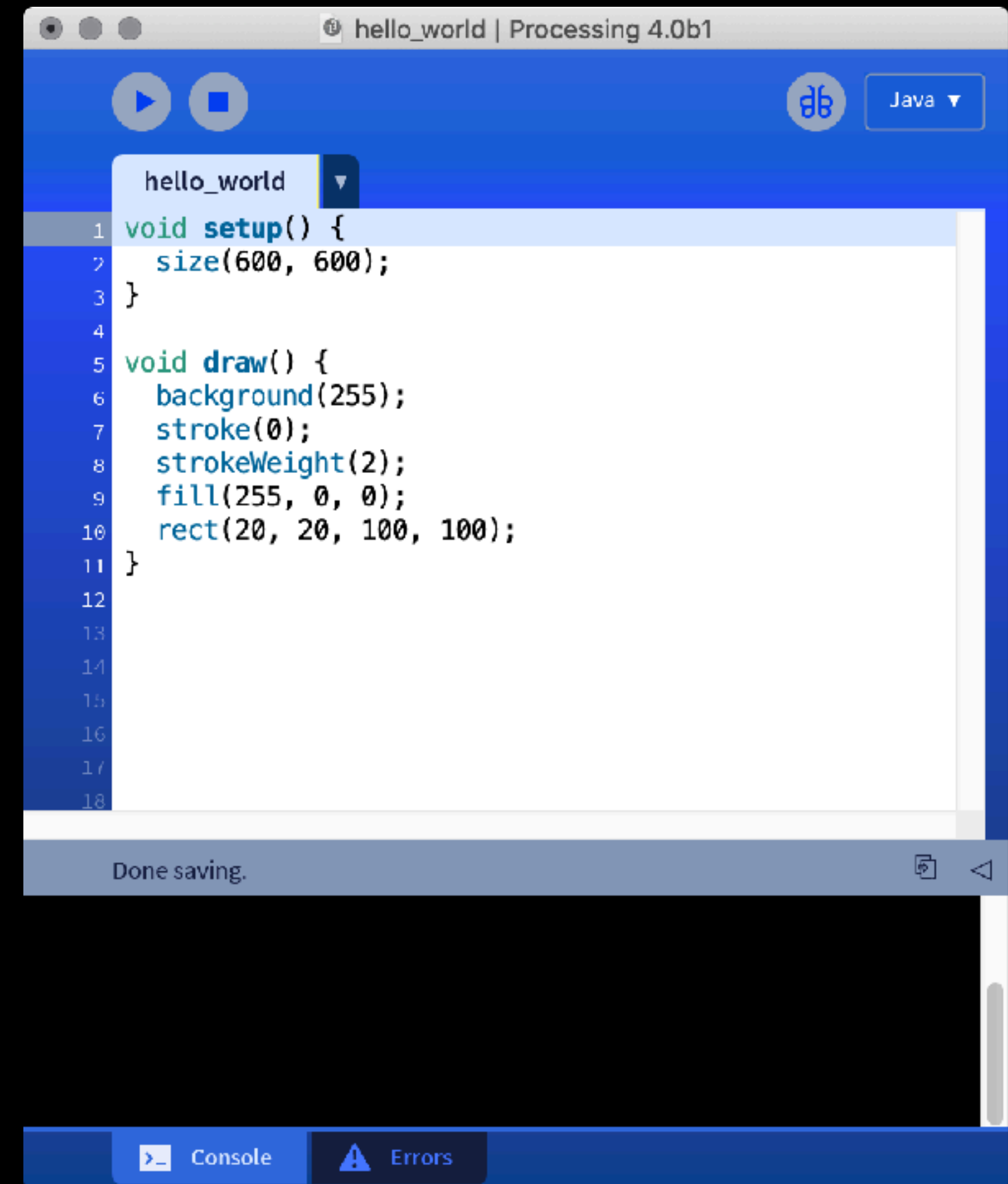


# Stroke & Fill

*what does this code do?*

```
stroke(0);  
strokeWeight(2);
```

```
fill(255,0,0);
```



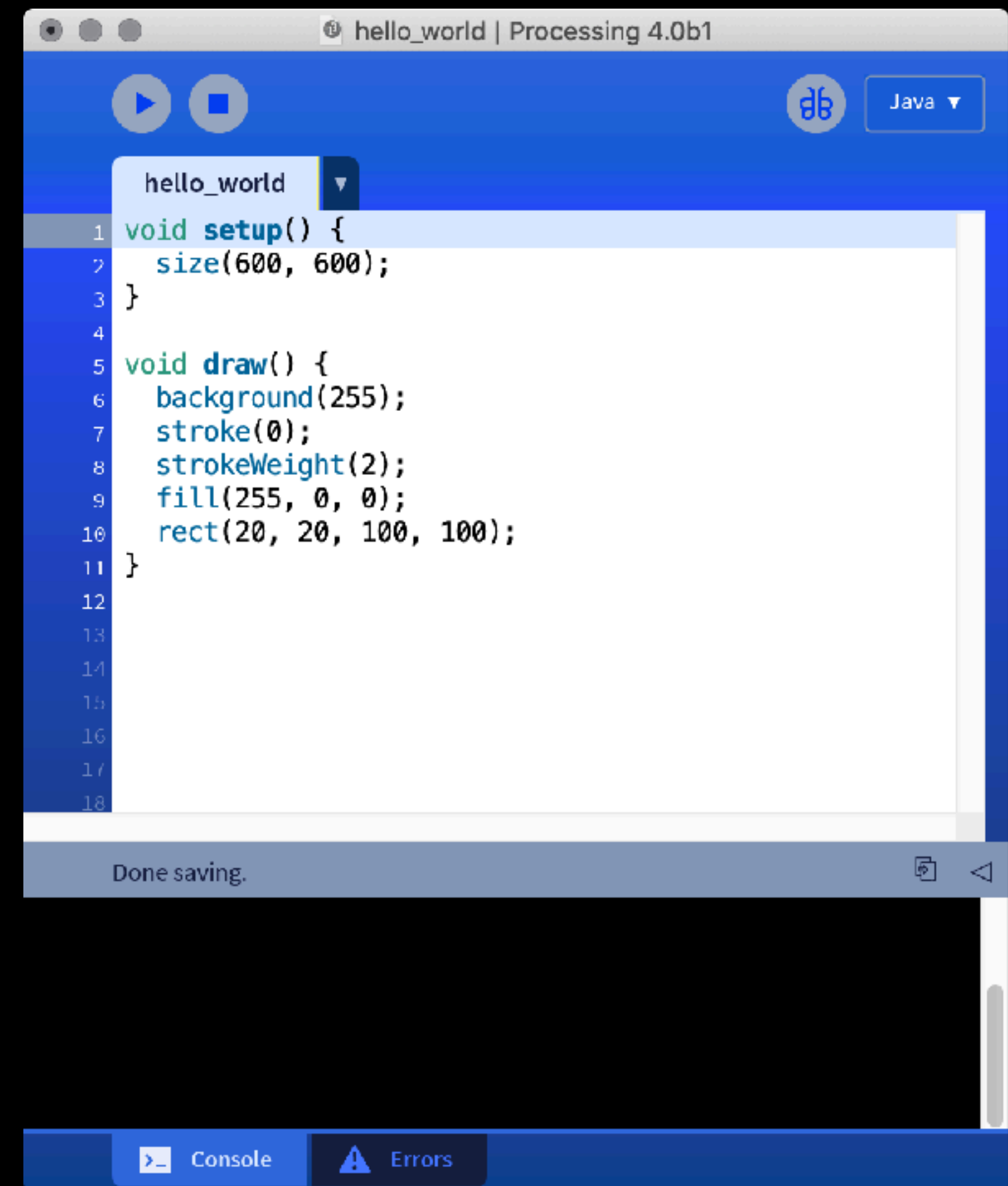
# Stroke & Fill

*what does this code do?*

```
stroke(0);  
strokeWeight(2);
```

```
fill(255,0,0);
```

*It gives the rectangle a thicker stroke, sets it to black, and fills the square red.*



# Challenge #2

Give all your objects different stroke and fill colors and weights.

Tip: Look at the `stroke`, `strokeWeight`, and `fill` functions in the documentation.

# Questions

How do we change the stroke or fill of shapes like in Illustrator?

What color is defined by RGB components (255, 0, 255)?

What if you want a shape to have a fill but no stroke?

How would you make the fill color the same for multiple shapes?

# Challenge #3

Write a function that saves a frame as a png file when you press a key.

Tip: Look at the docs for `saveFrame()` and `keyPressed()`.

# Questions

Where does the file get saved when you press your save key?

How do you give each save instance a unique filename?

How do you save your sketch as a pdf?

*(Tip: Look at the docs. Notice a pattern here?)*

# Where to next?

Continue learning from tutorials and examples and by reading the reference documentation.

Watch all the entry level tutorials and work at writing code. The best way to learn how to write code is to write code. Use comments everywhere to make notes.

Watch Coding Train videos. They are excellent.

Pick up a copy of Learning Processing to dive right in.

Start a Processing work study group. Make things together.

Look for opportunities to use this amazing tool in your workflow.

BREAK



*EVERYDAY*

*How does one go about  
doing an everyday or daily  
project?*

*Constraints*

*Prompts*

*Verbs*

*Determination*

*Why challenge yourself  
with every day projects?*

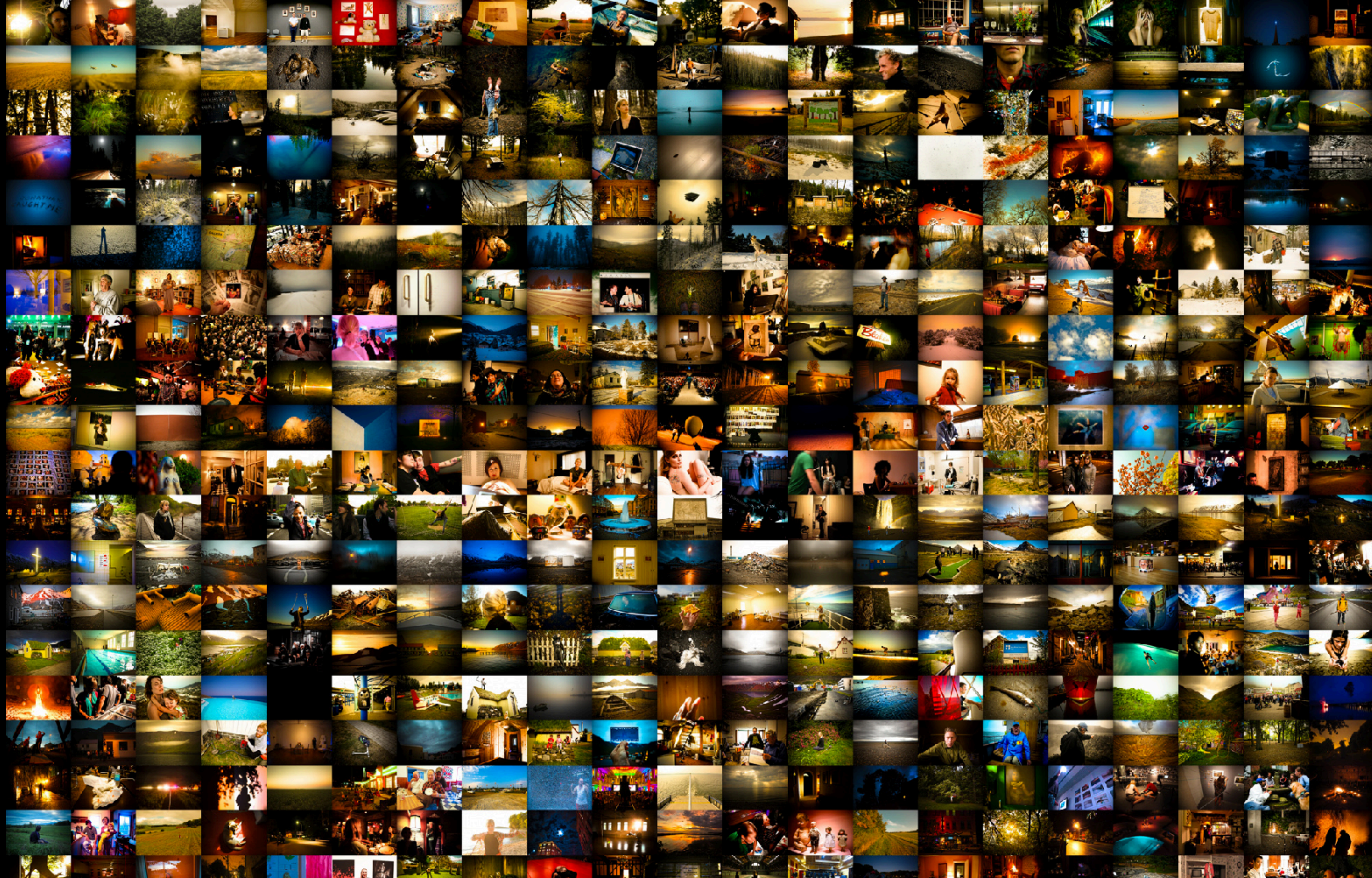
*Self Improvement*

*Leveling up your skillset*

*Becoming a better human*

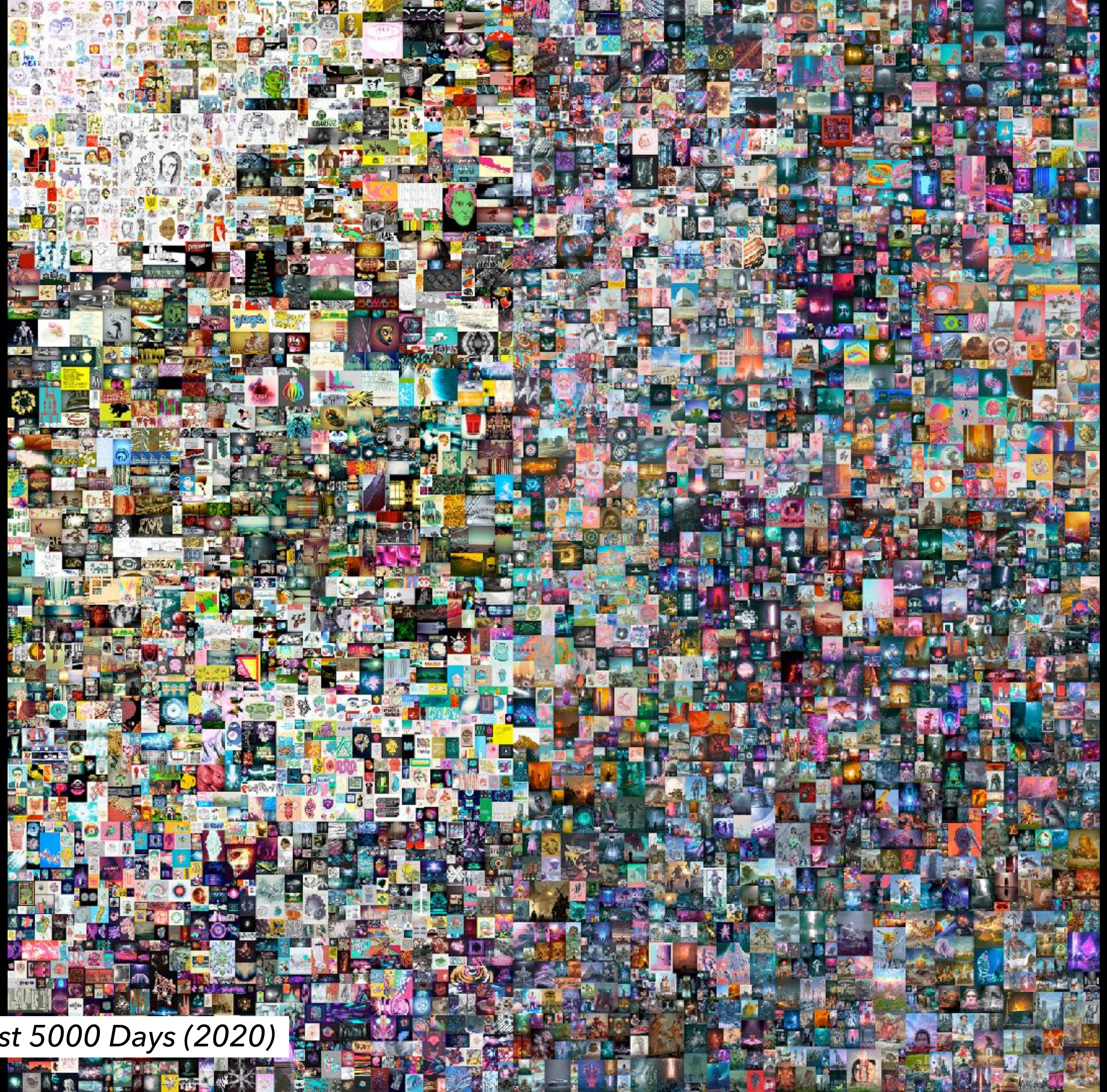
*Creating the reality you want*





Jonathan Harris - Today (2010)



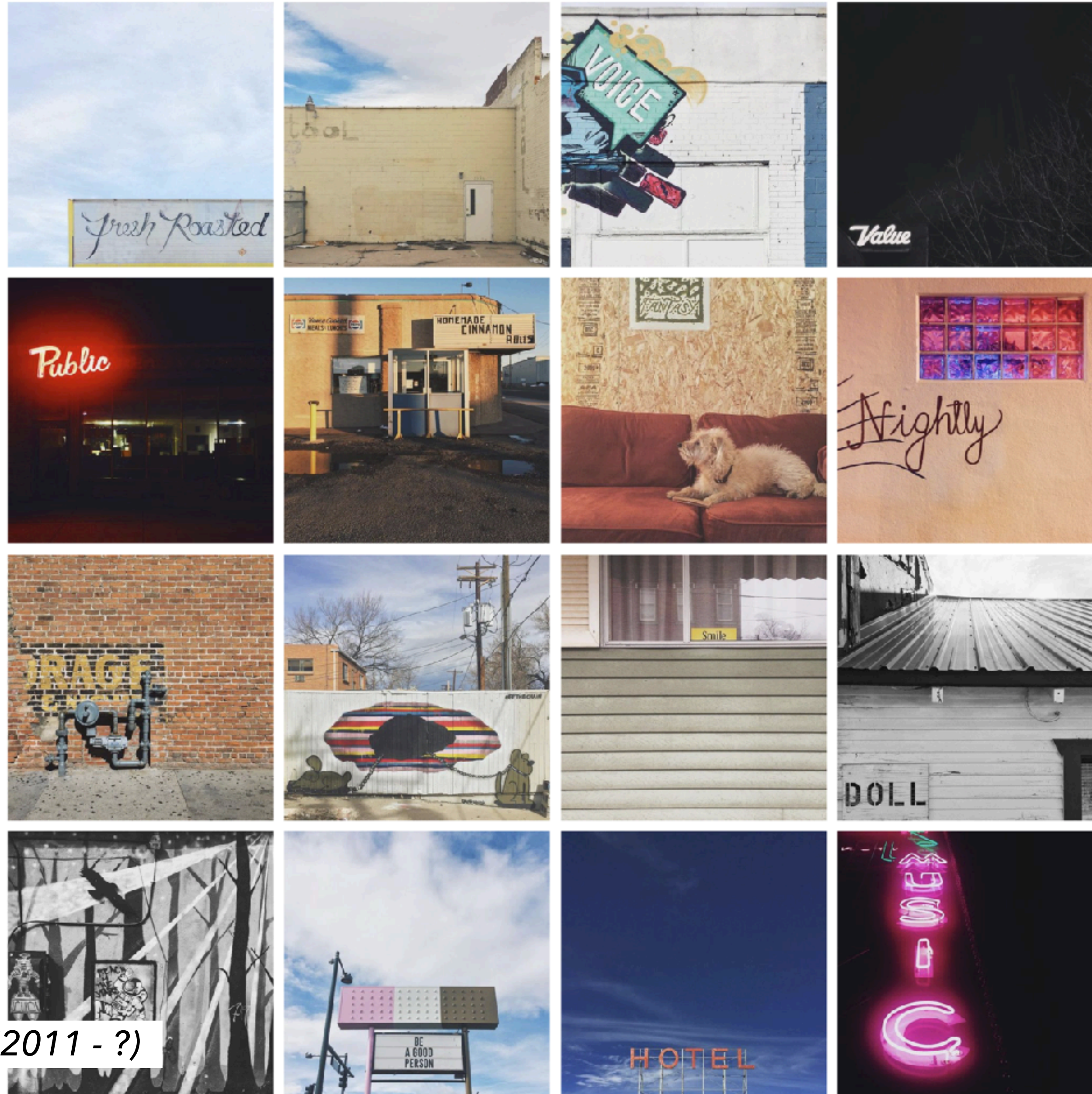


*Beeple - Everyday: The First 5000 Days (2020)*



# Collecting words

[Home](#) [About](#) [Buy the book](#) [Living room](#) [Contact](#) [Archive](#) [Submit a post](#) [Q](#)



Brian Fouhy - Collecting Words (2011 - ?)





## 365 Days of OP-1

01/2016

Launch Project

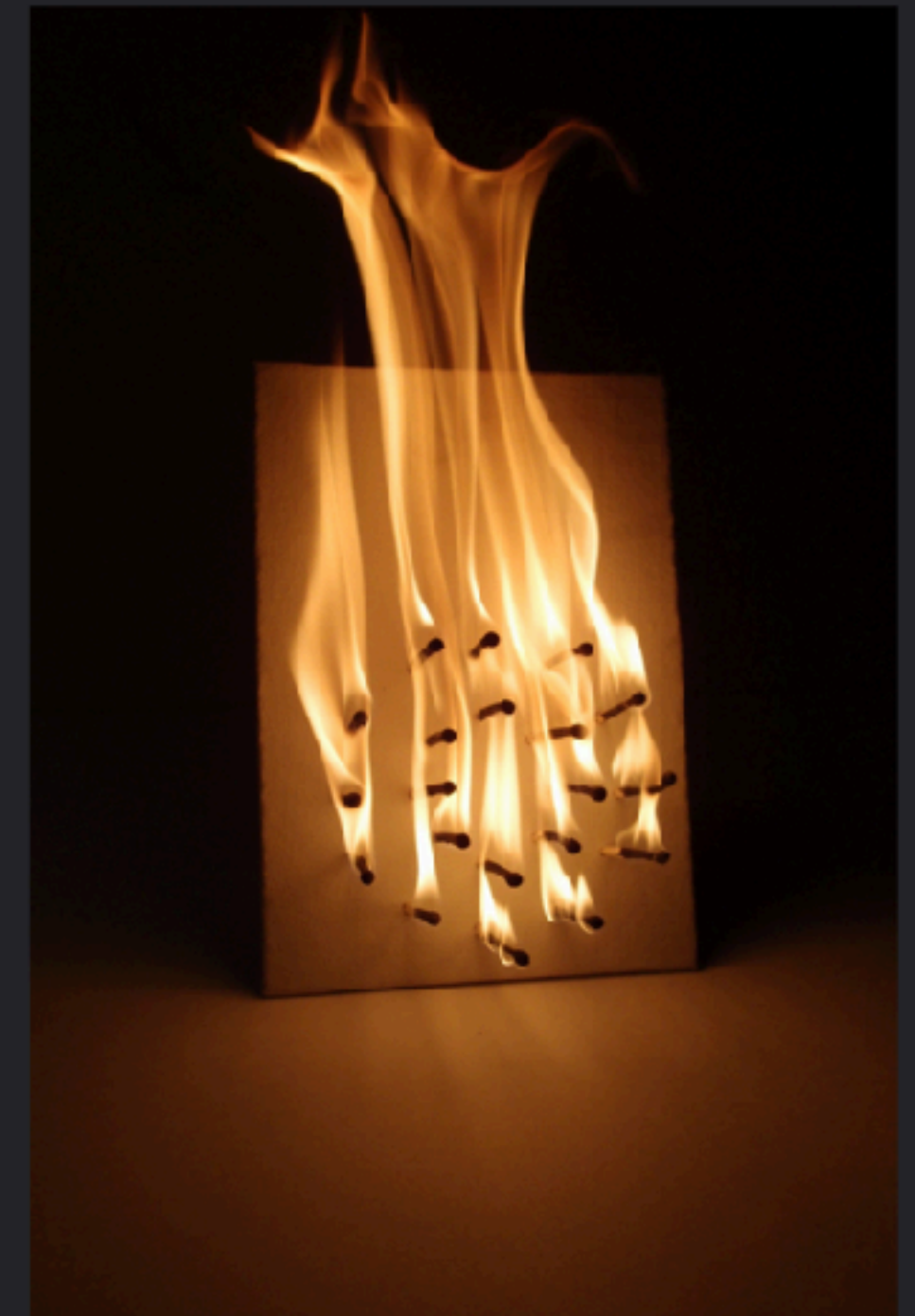
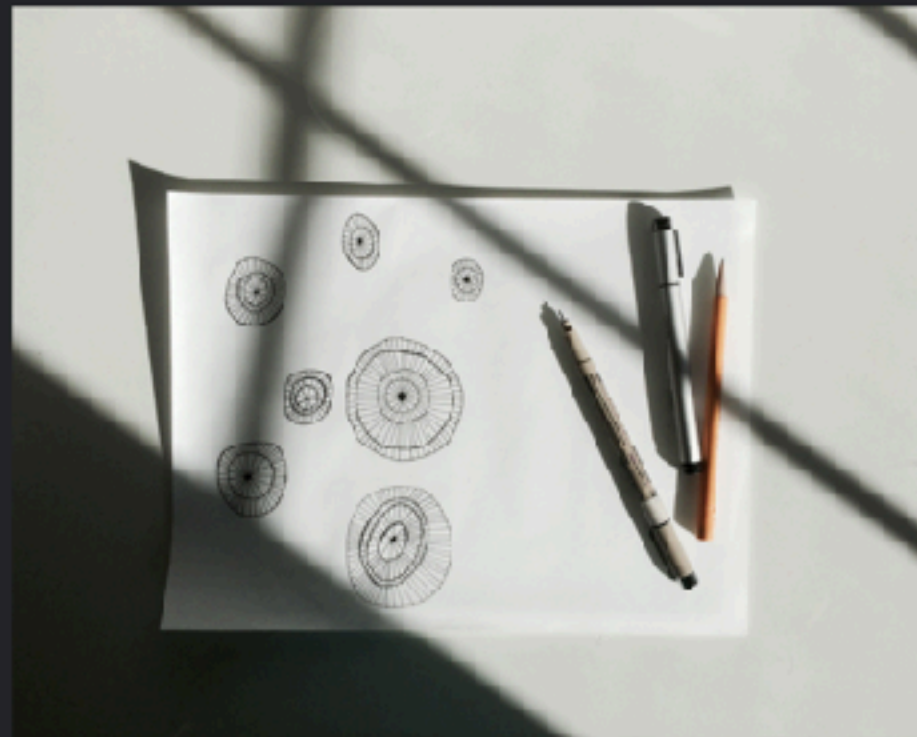
CW&T - 365 Days of OP-1 (2016)



## SIGNIFICANT NONSENSE

daily design doodles by  
chelsea lee

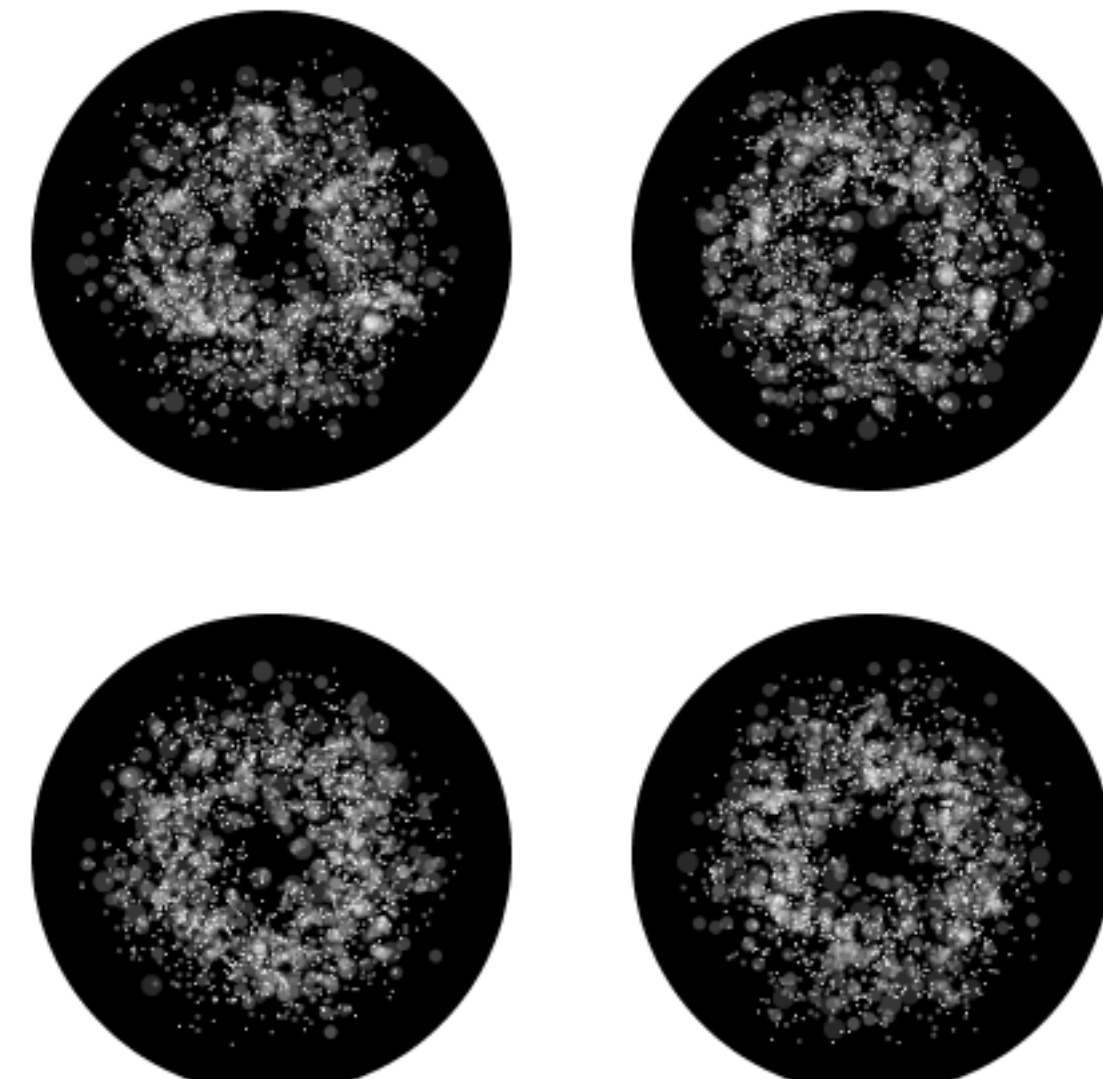
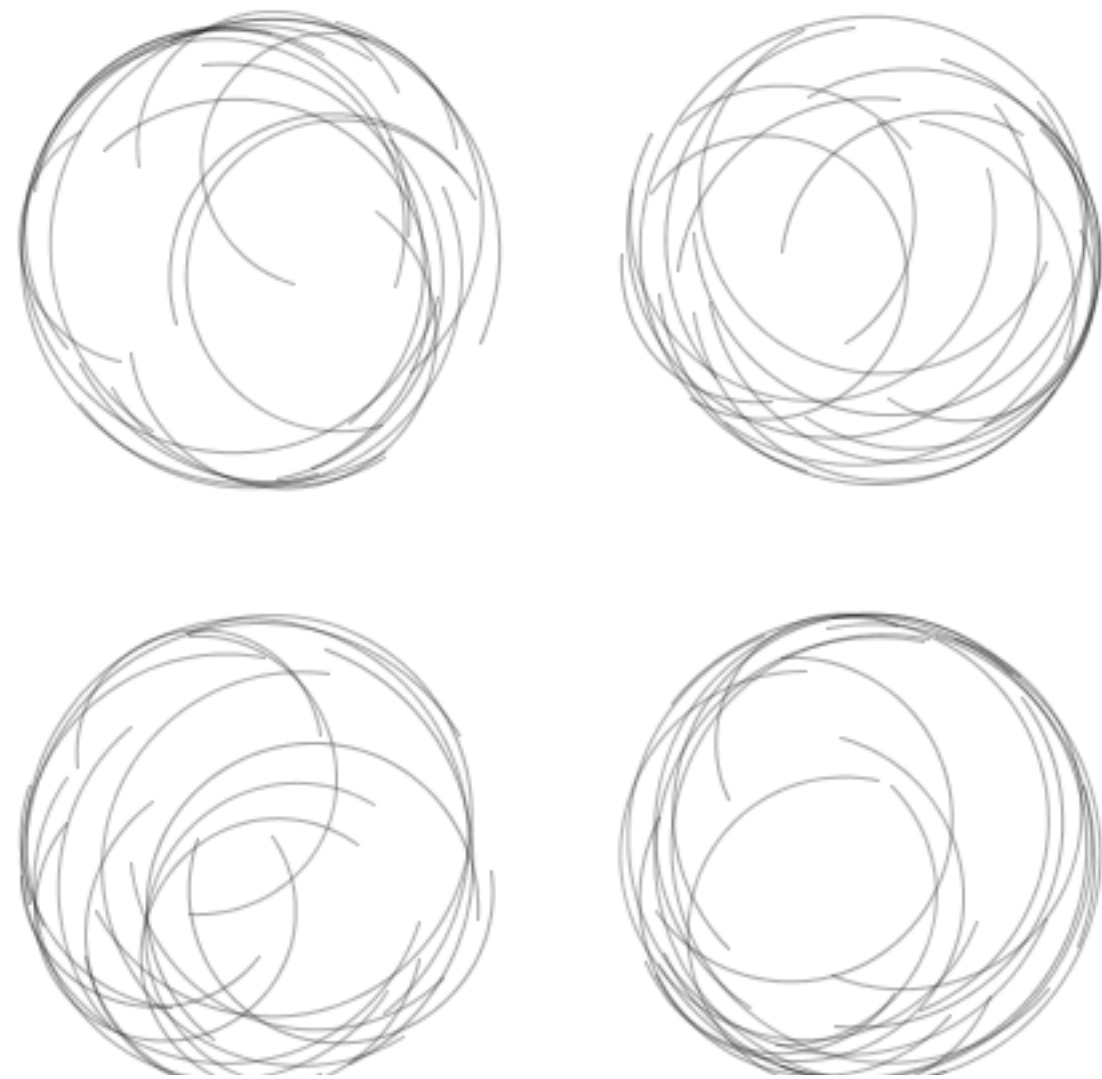
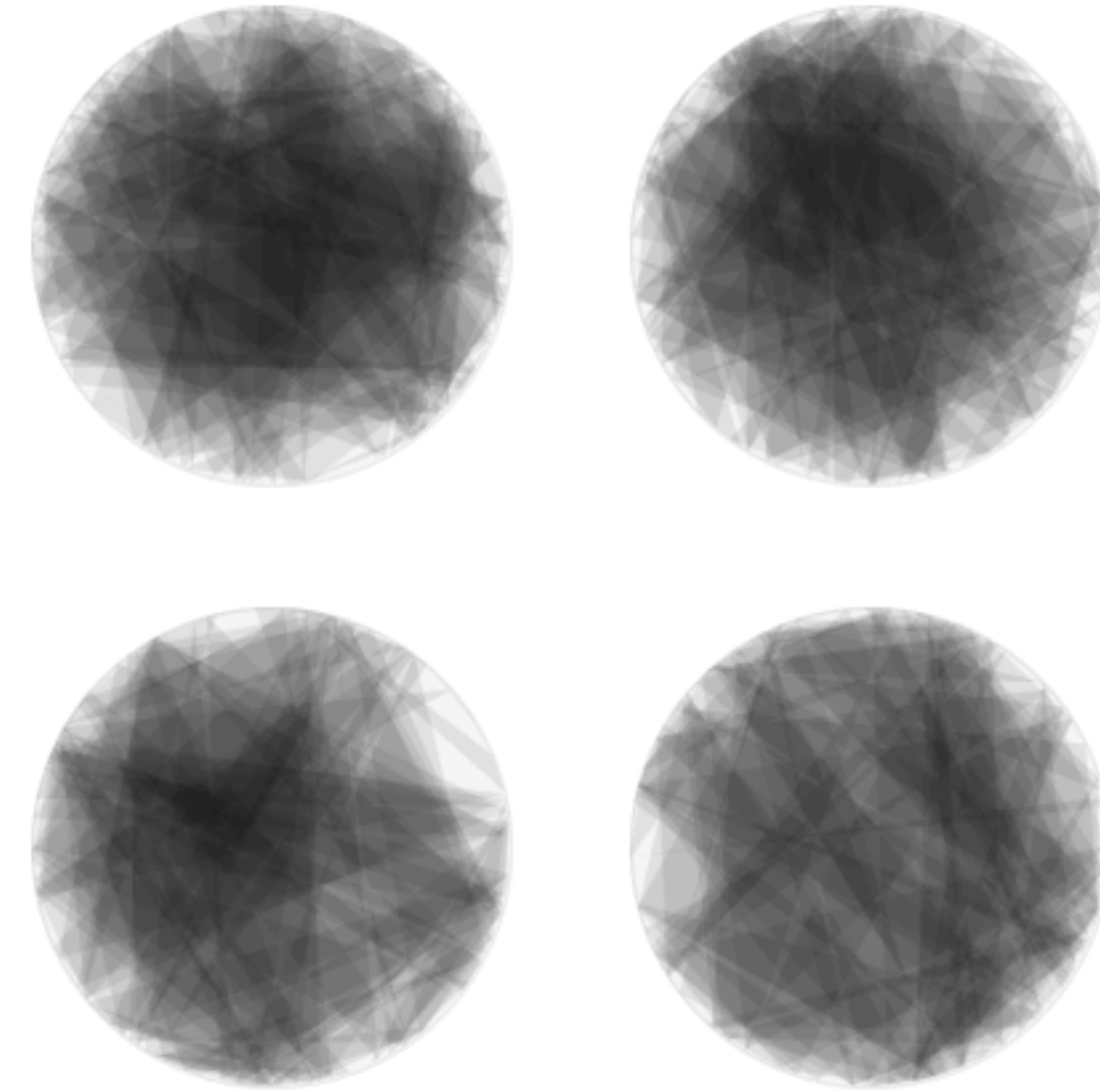
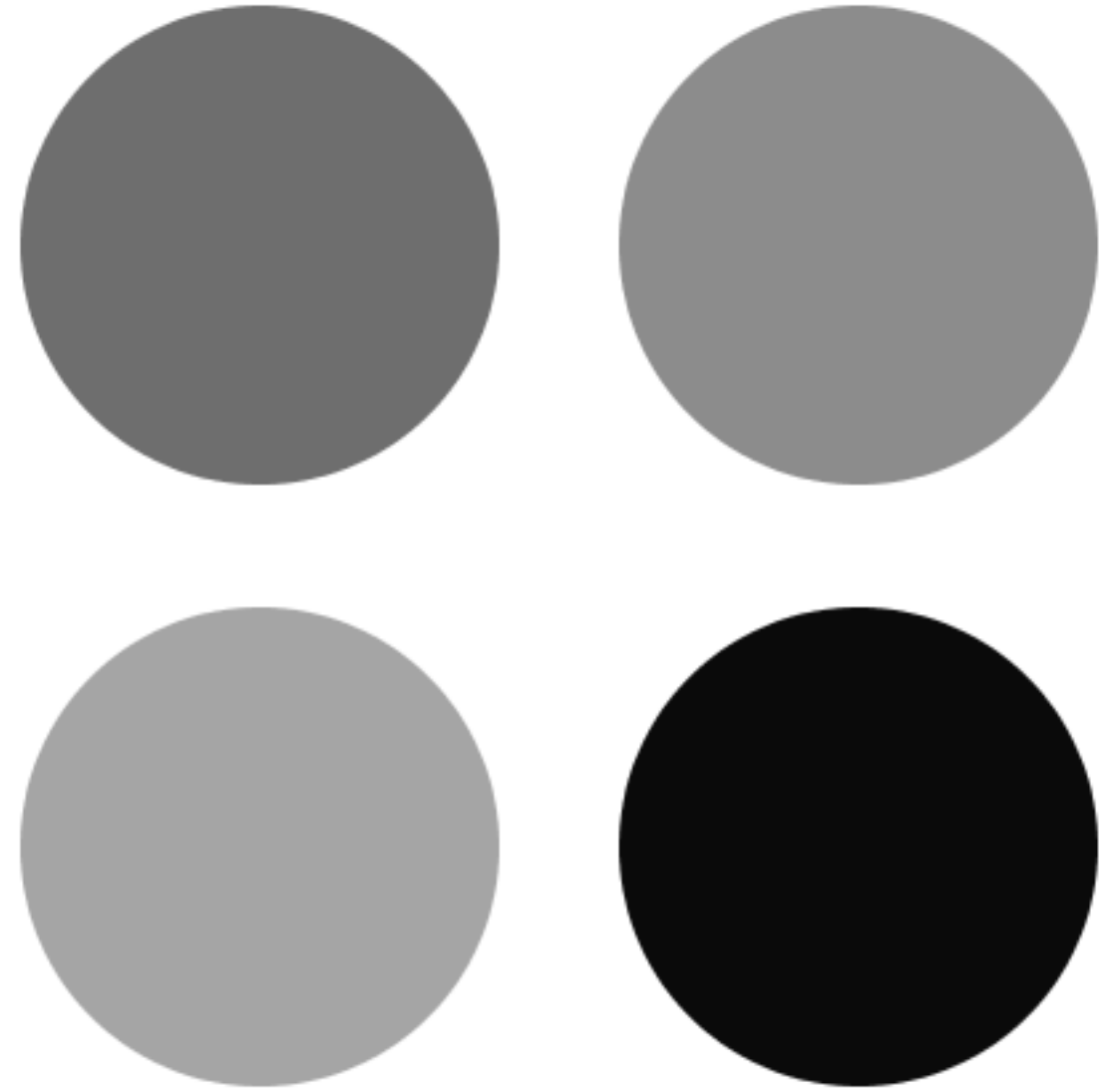
JANUARY  
FEBRUARY  
MARCH  
APRIL  
MAY  
JUNE  
JULY  
AUGUST  
SEPTEMBER  
OCTOBER  
NOVEMBER  
DECEMBER



Significant Nonsense (2011)

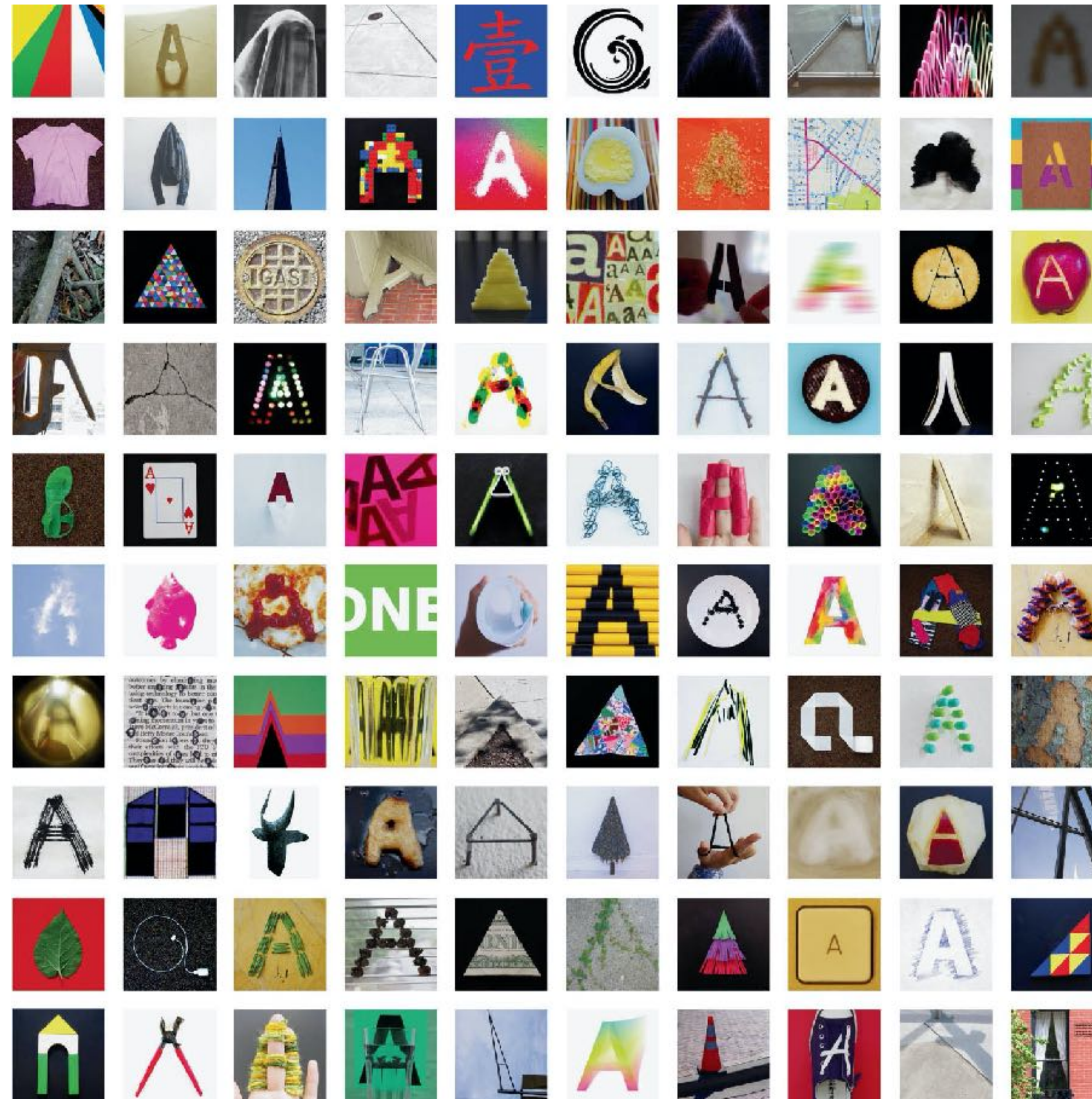
Constraints:  
working daily with type,  
sketching, ui/ux, experimental  
art





*Constraints:  
75 instances, no  
color, modularity  
(functions can be  
reused), generative  
or procedural (each  
call to a function  
returns different  
result)*





*Artwork by Yingxi Zhou. One Hundred Iterations, Graphic Design: The New Basics.*



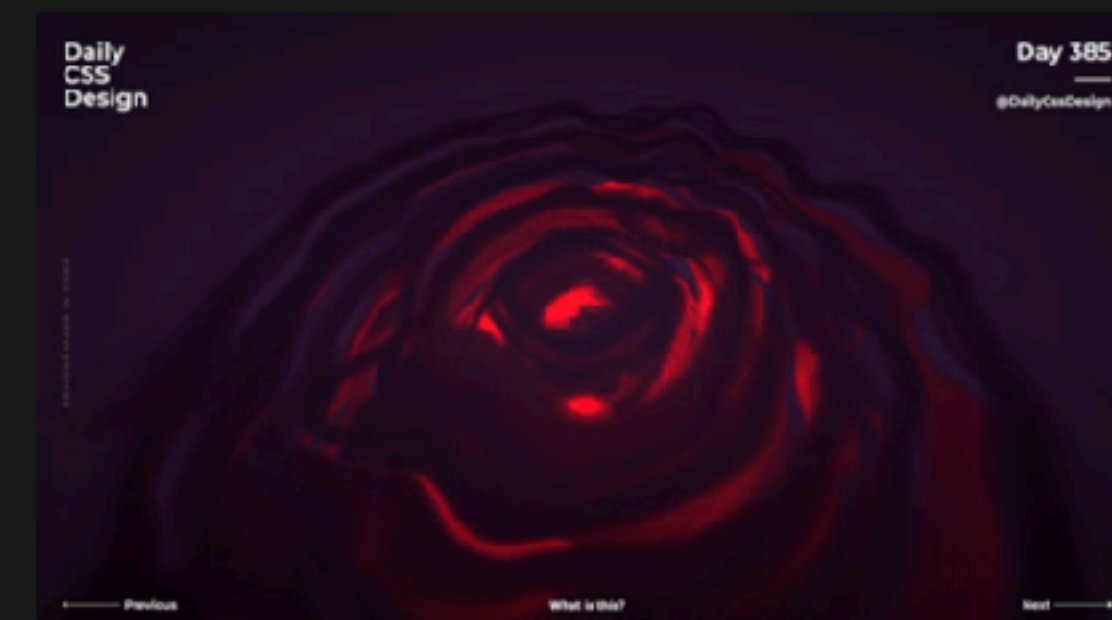
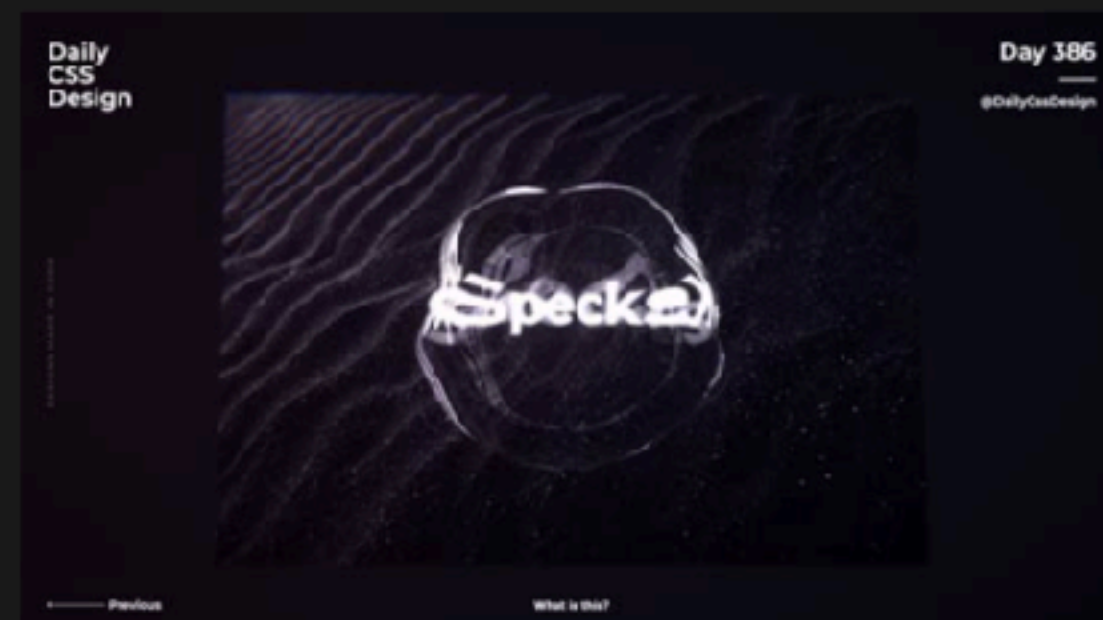


*Eggs (2013), Studio Courses by Andrew Kudless*



# Hi! I'm a web designer and this is my side project

The goal was to create one design every day for a year. All shapes, patterns and colors are made by coding. [Read more](#)





# GENUARY 2021

```
10 INPUT PROMPT
20 WRITE CODE
30 GENERATE
40 POST #GENUARY #GENUARY2021
50 GOTO 10
```

## GENUARY?

GENUARY is an artificially generated month of time where we build code that makes beautiful things.

It happens during the month of January 2021.

For every 24 hour day within this 744 hour timespan, we have prepared a prompt with instructions for you to execute.

You don't have to follow the prompt exactly. Or even at all. But, y'know, we put effort into this.

You can use any language, framework or medium. Please respect the Geneva Conventions.

Share your work and tag it with **#genuary** and **#genuary2021**. If you're going to misspell it as #genaury, please go all in and include all permutations.

01.

02.

03.

04.

05.

06.

07.

08.

09.

10.

11.

12.

13.

14.

15.

16.

17.

18.

19.

20.

21.

22.

23.

24.

25.

26.

27.

28.

29.

30.



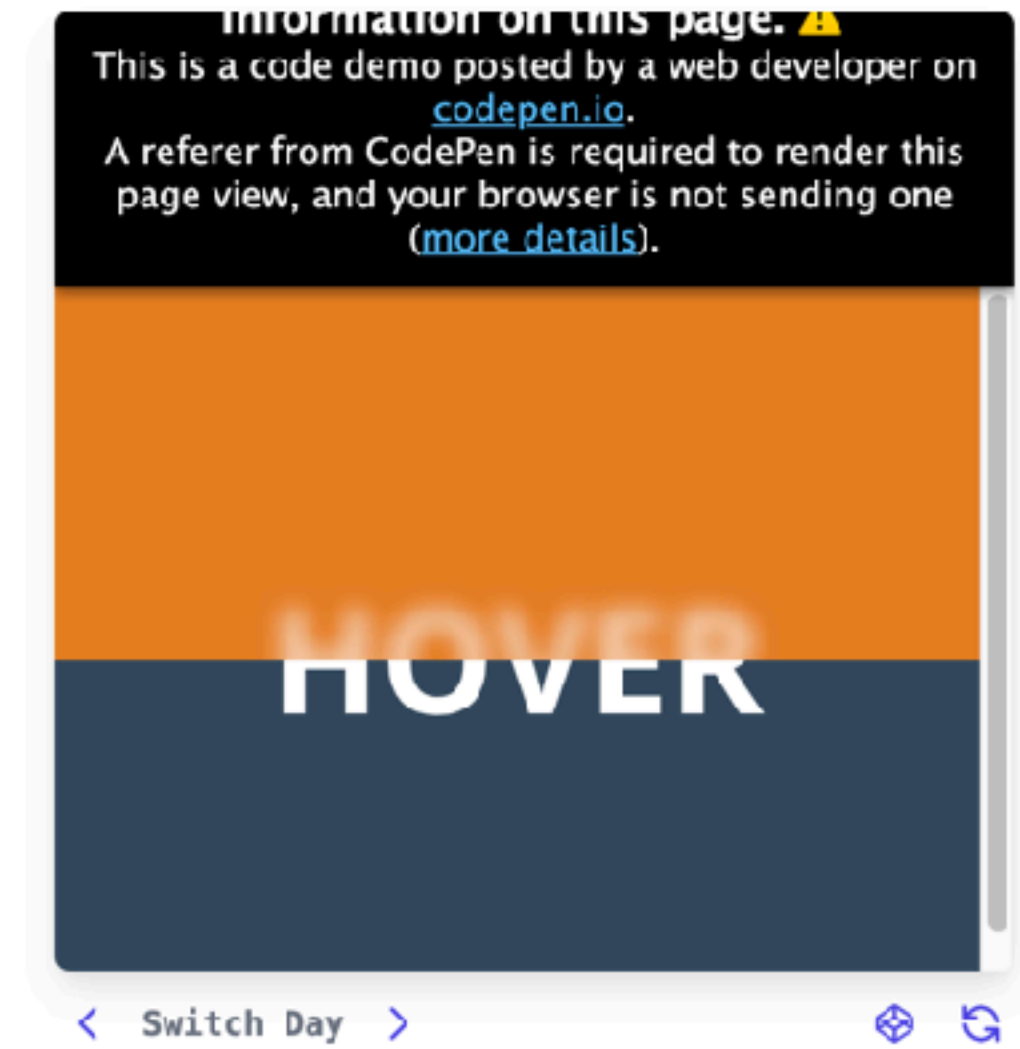
## 100 DAYS CSS CHALLENGE

## Day #60

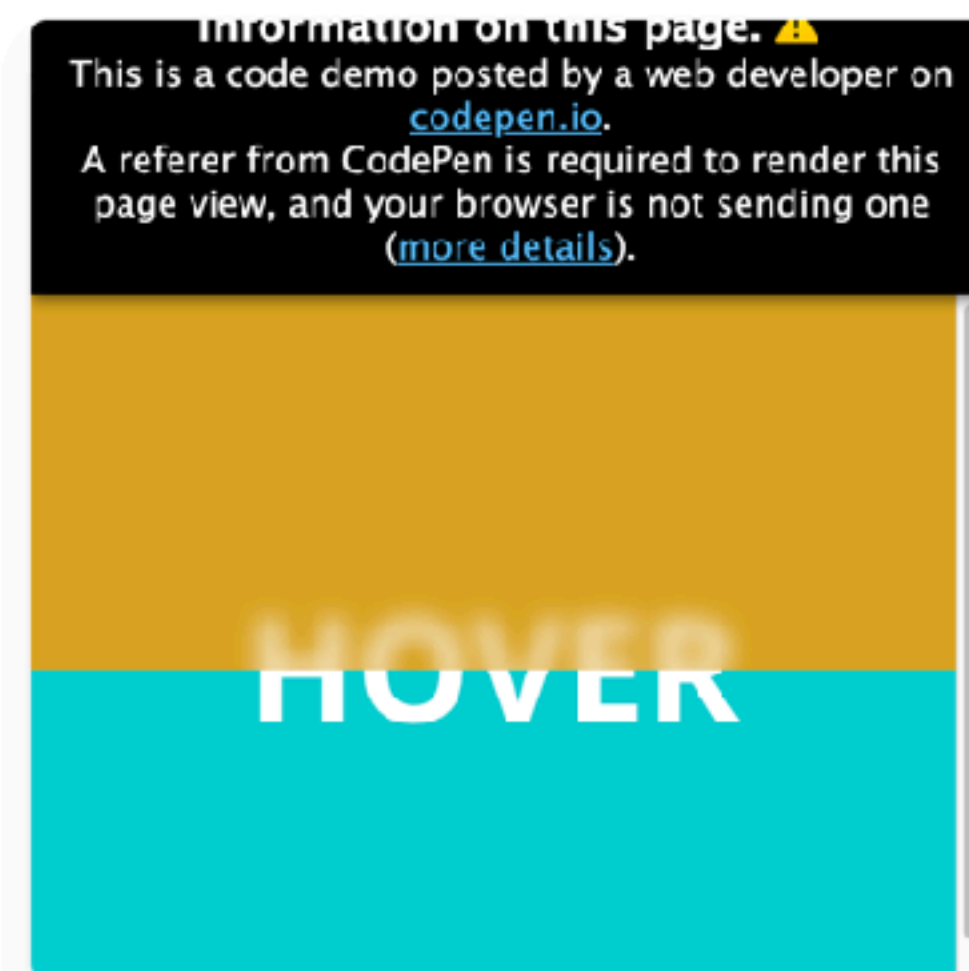
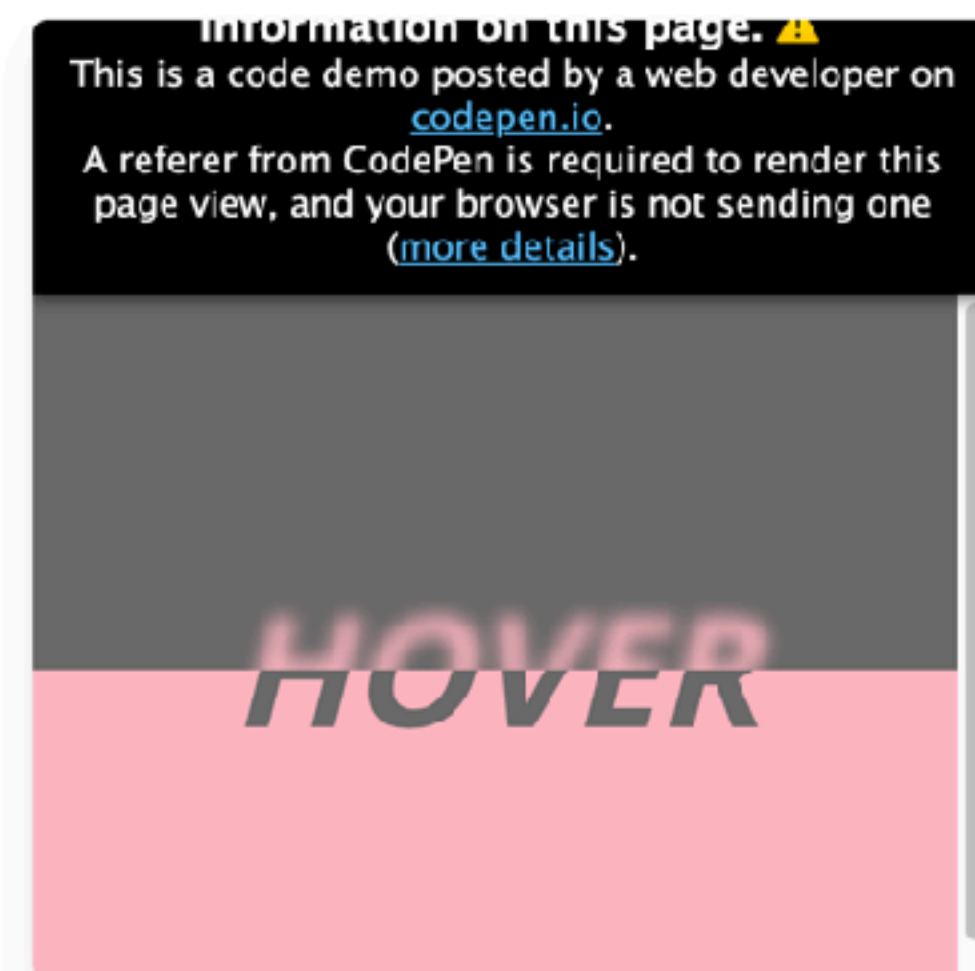
**Blurry Overlay:** Backdrop blur effects are very trendy, but they are difficult to implement with CSS without any tricks.

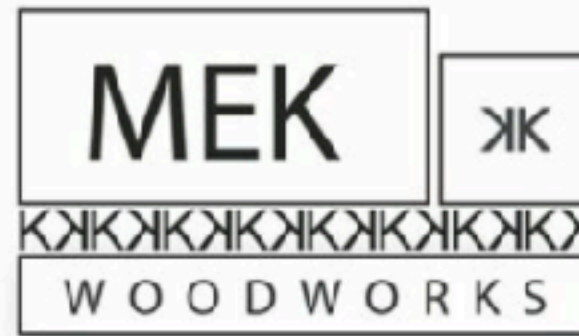
» Grab the [> TEMPLATE](#) and submit your work.

⌚ 08:25:56



## SUBMITTED PENS





[Home](#) [52 Boxes](#) [Portfolio](#) [About](#) [Blog](#) [Teaching](#) [Store](#) [Log In](#) [Cart \(0\)](#)

## 52 BOXES IN 52 WEEKS

Here they are. Every box I've made in my 52 weeks, 52 boxes challenge. What's that you ask? In attempt to develop my own design aesthetic, I'm made 52 boxes over the course of one year. Want to know more about what it's all about? [Read this](#). Want to know more about one of the boxes? Click the box's picture. Want to to buy one of them? Send [me](#) an email. Want to send me a gift basket of rum, beer, and Swedish fish? Please do. Or just go [buy the book](#).



Box 1



Box 2





to roll  
to crease  
to fold  
to store  
to bend  
to shorten  
to twist  
to dapple  
to crumple  
to shave  
to tear  
to chip  
to split  
to cut  
to sever  
to drop  
to remove  
to simplify  
to differ  
to disarrange  
to open  
to mix  
to splash  
to knot  
to spill  
to droop  
to flow

to curve  
to lift  
to inlay  
to impress  
to fore  
to flood  
to smear  
to rotate  
to swirl  
to support  
to hook  
to suspend  
to spread  
to hang  
to collect  
of tension  
off gravity  
of entropy  
of nature  
of grouping  
of layering  
of felting  
to grasp  
to tighten  
to bundle  
to heap  
to gather

to scatter  
to arrange  
to repair  
to discard  
to pair  
to distribute  
to surfeit  
to complement  
to enclose  
to surround  
to encircle  
to hide  
to cover  
to wrap  
to dig  
to tile  
to bind  
to weave  
to join  
to match  
to laminate  
to bond  
to hinge  
to mark  
to expand  
to dilute  
to light

to modulate  
to distill  
of waves  
of electromagnetic  
of inertia  
of ionization  
of polarization  
of refraction  
of simultaneity  
of tides  
of reflection  
of equilibrium  
of symmetry  
of friction  
to stretch  
to bounce  
to erase  
to spray  
to systematize  
to refer  
to force  
of mapping  
of location  
of context  
of time  
of carbonization  
to continue



RJ Duran  
rjduran.net  
rjduranjr@gmail.com  
970.214.3072



# ASSIGNMENT 1

*Assignment 0:*  
*Complete this by class Monday*

Student Questionnaire



*QUESTIONS ?*