

py5 for novel data visualizations using python

//vuw-snap 2022-04-12

./processing

processing

a graphical library and i.d.e built for the electronic arts, new media art, and visual design communities with a focus on teaching non-programmers the fundamentals of computer programming in a visual context (using java)

but the community has grown to include tens of thousands of students, artists, designers, researchers, and hobbyists who use processing for learning and prototyping

processing includes variants for javascript, python, ruby, scala, clojure, and more

Welcome to Processing! / Processing.org — Mozilla Firefox

Welcome to Processing! / Pr X

Processing Foundation

Processing p5.js Processing Android Processing Python

Processing Download Documentation Learn Teach About Donate Search

Welcome to 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.



https://processing.org

The image shows two windows from the Processing 4.0b5 IDE. The left window is titled "testPythonProc4 | Processing 4.0b5" and contains Python code. The right window is titled "testPythonProc4" and displays the resulting graphical output.

Code (testPythonProc4.py):

```
1 def setup():
2     size(300, 200)
3
4
5 def draw():
6     background(255)
7     fill(255, 0, 0)
8     circle(width/2, height/2, 100)
9     text("this is a string", width/2, 50)
10
11
12
13
```

Output (testPythonProc4):

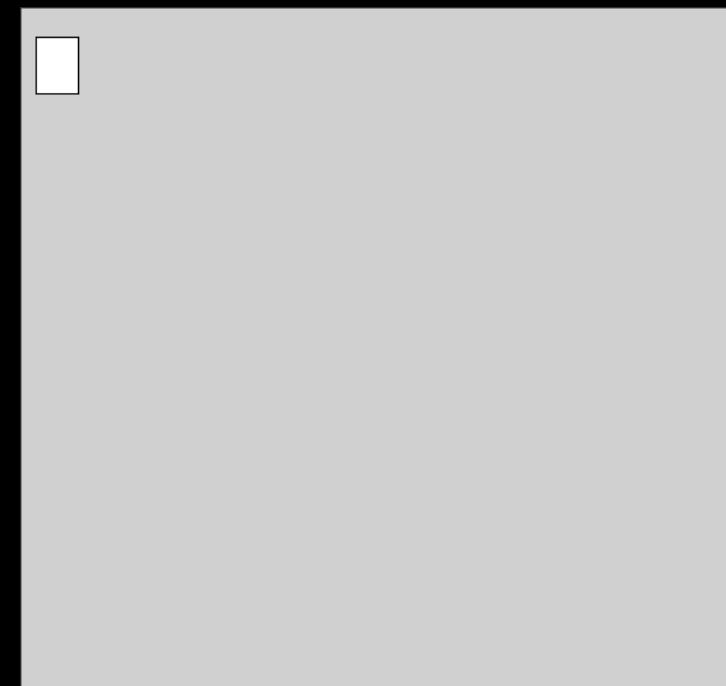
A red circle is centered at the coordinates (150, 100). Above the circle, the text "this is a string" is displayed in red font.

Bottom Right: Submit new issue

<https://processing.org>

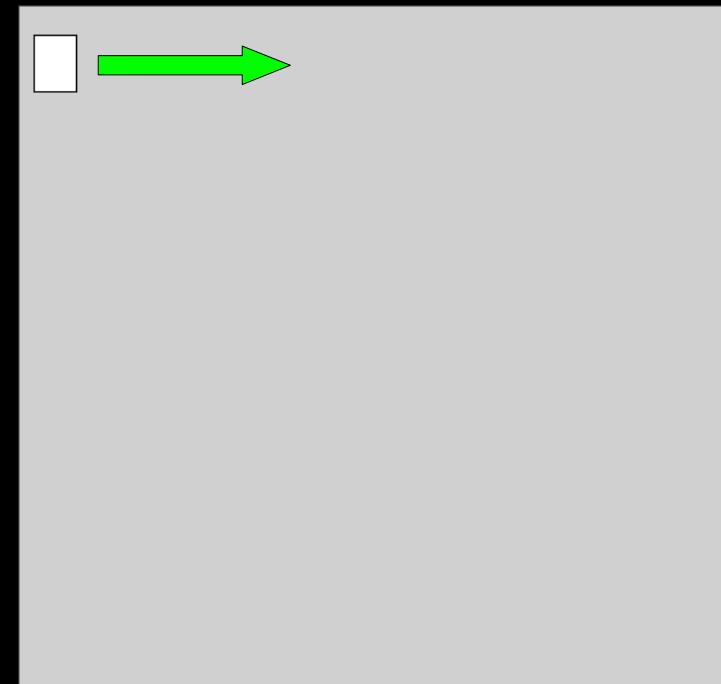
./processing

```
size(500, 500);  
rect(10, 20, 30, 40);
```



./processing

```
void setup() {  
    size(500, 500);  
}  
  
void draw() {  
    rect(second(), 20, 30, 40);  
}
```



./processing_work

processing work

Fidenza by Tyler Hobbs - Collection | OpenSea — Mozilla Firefox

OpenSea

Search items, collections, and accounts

Explore Stats Resources Create

Filter

Status

Buy Now On Auction

New Has Offers

Buy with Card

Price

United States Dollar (USD)

Min to Max

Apply

Fidenza by Tyler Hobbs - Col X

https://opensea.io/collection/fidenza-by-tyler-hobbs

Fidenza by Tyler Hobbs - Collection | OpenSea — Mozilla Firefox

Fidenza #91

Fidenza #863

Fidenza #560

Price

Price

Price

Last

7 days left

6 days left

217

14

89

51.9

52

56

42.88

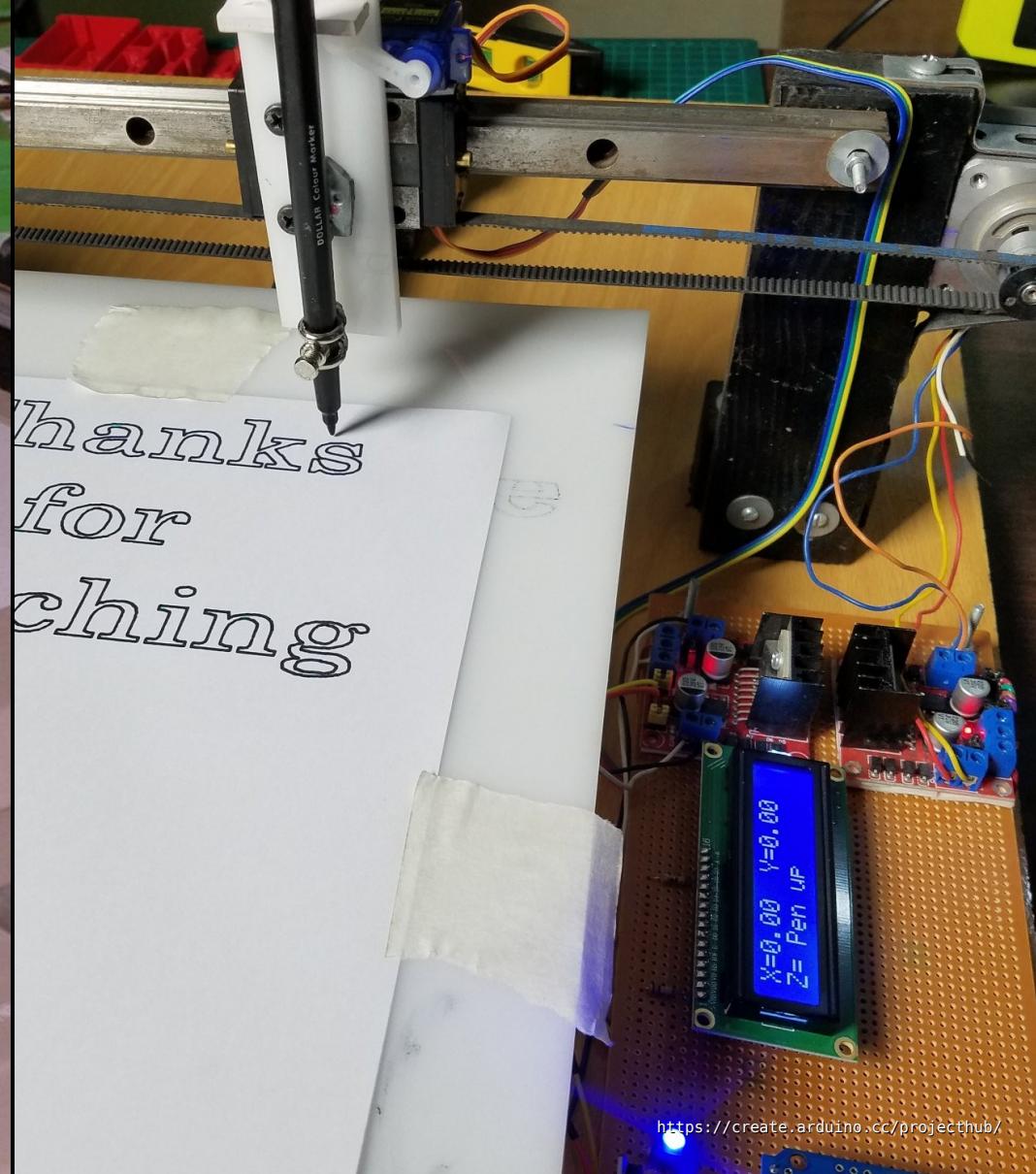
https://opensea.io/collection/fidenza-by-tyler-hobbs

The screenshot shows the OpenSea platform displaying the 'Fidenza by Tyler Hobbs' collection. The main content area shows four NFT items from the collection:

- Fidenza #91**: Price \$51.9, 7 days left.
- Fidenza #863**: Price \$52, 6 days left.
- Fidenza #560**: Price \$56, Last \$42.88.

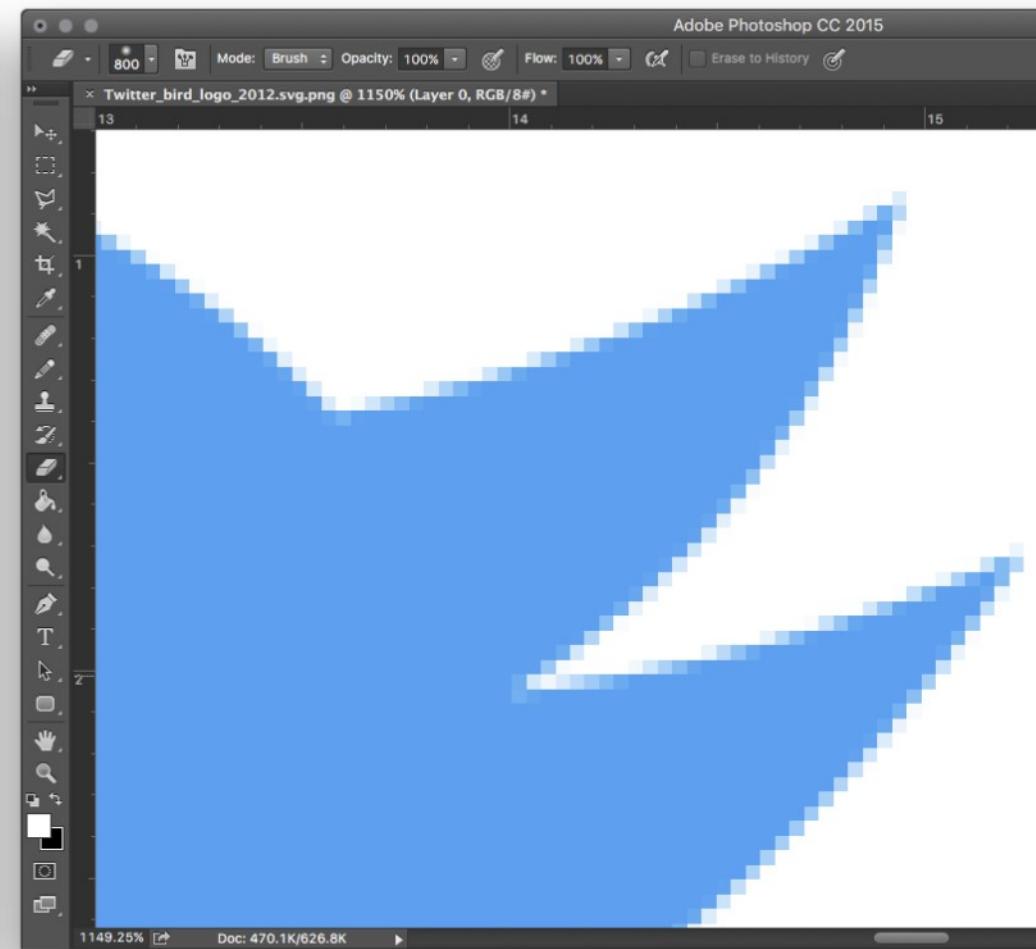
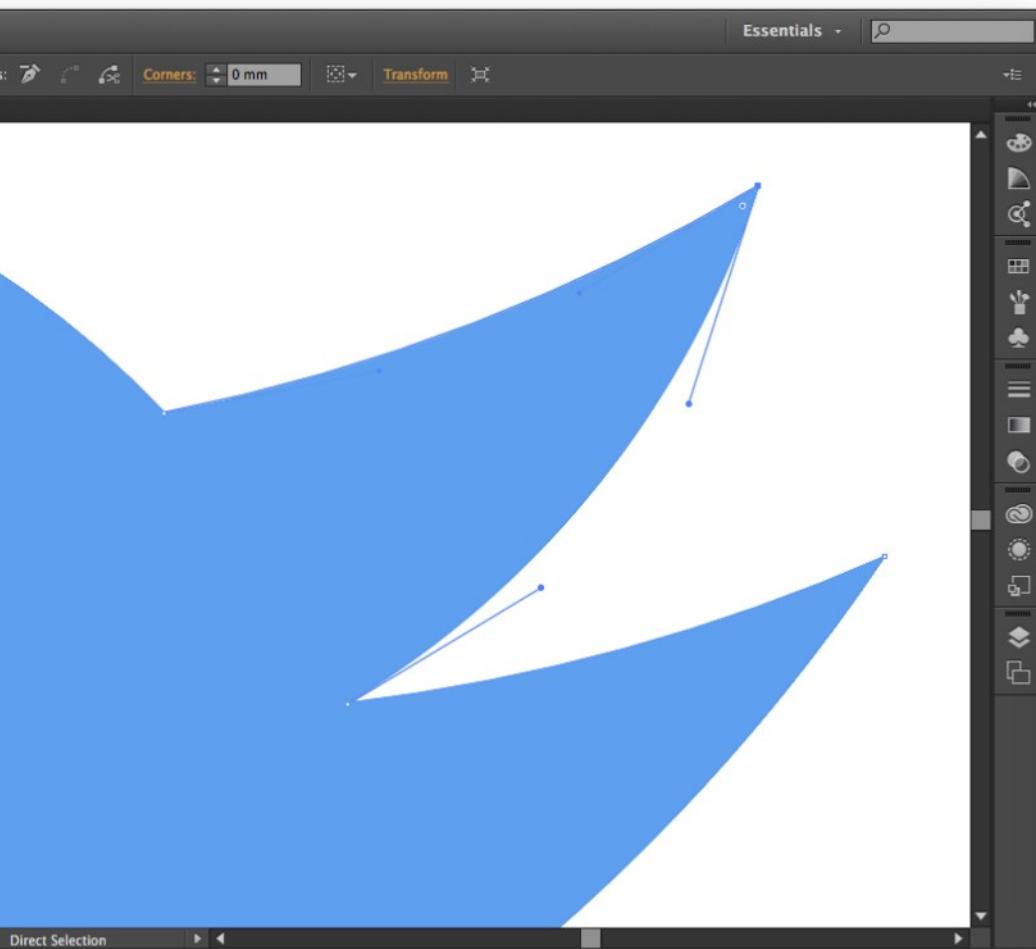
The sidebar on the left provides filtering and sorting options, including categories like 'Buy Now', 'On Auction', 'New', 'Has Offers', and 'Buy with Card'. It also includes a price filter section with a dropdown for 'United States Dollar (USD)' and input fields for 'Min' and 'Max' values, with an 'Apply' button at the bottom.

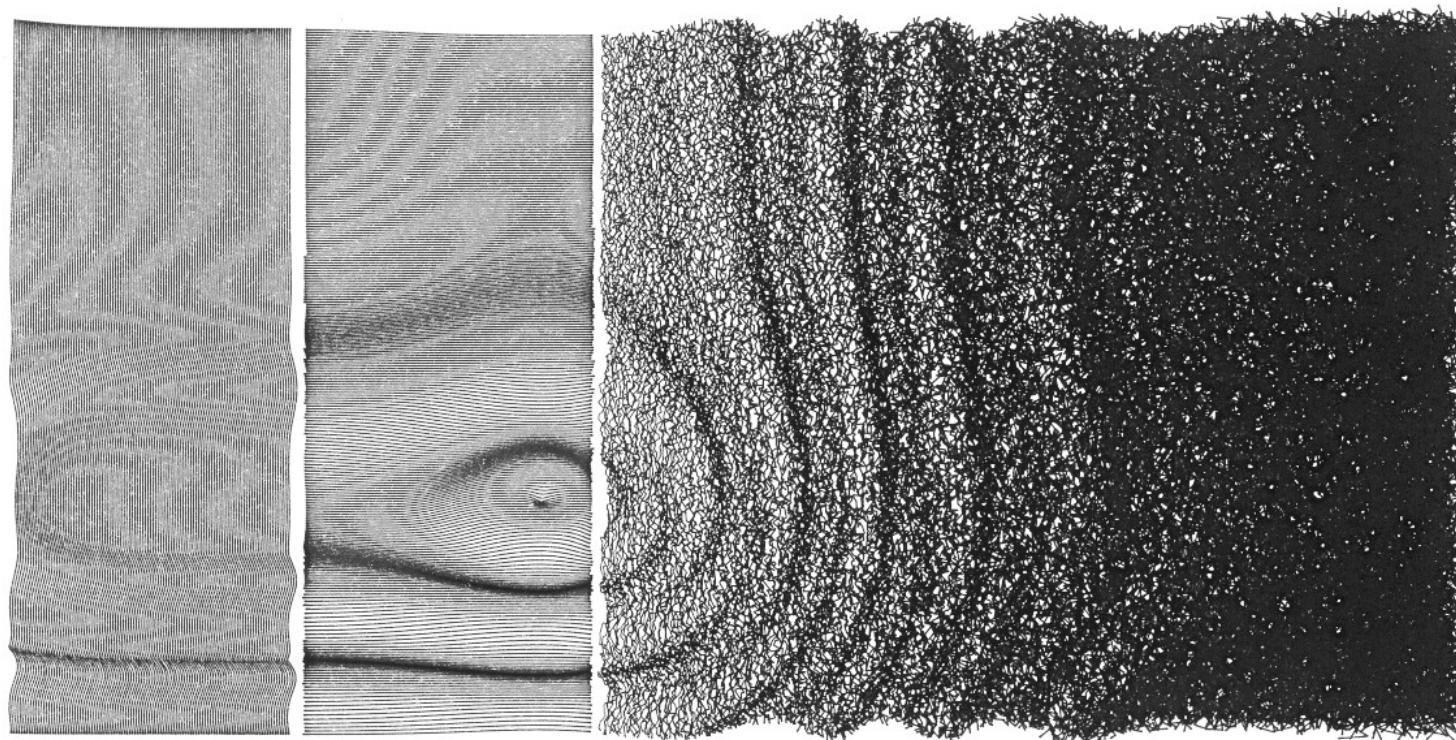
./processing_work/plotter_art



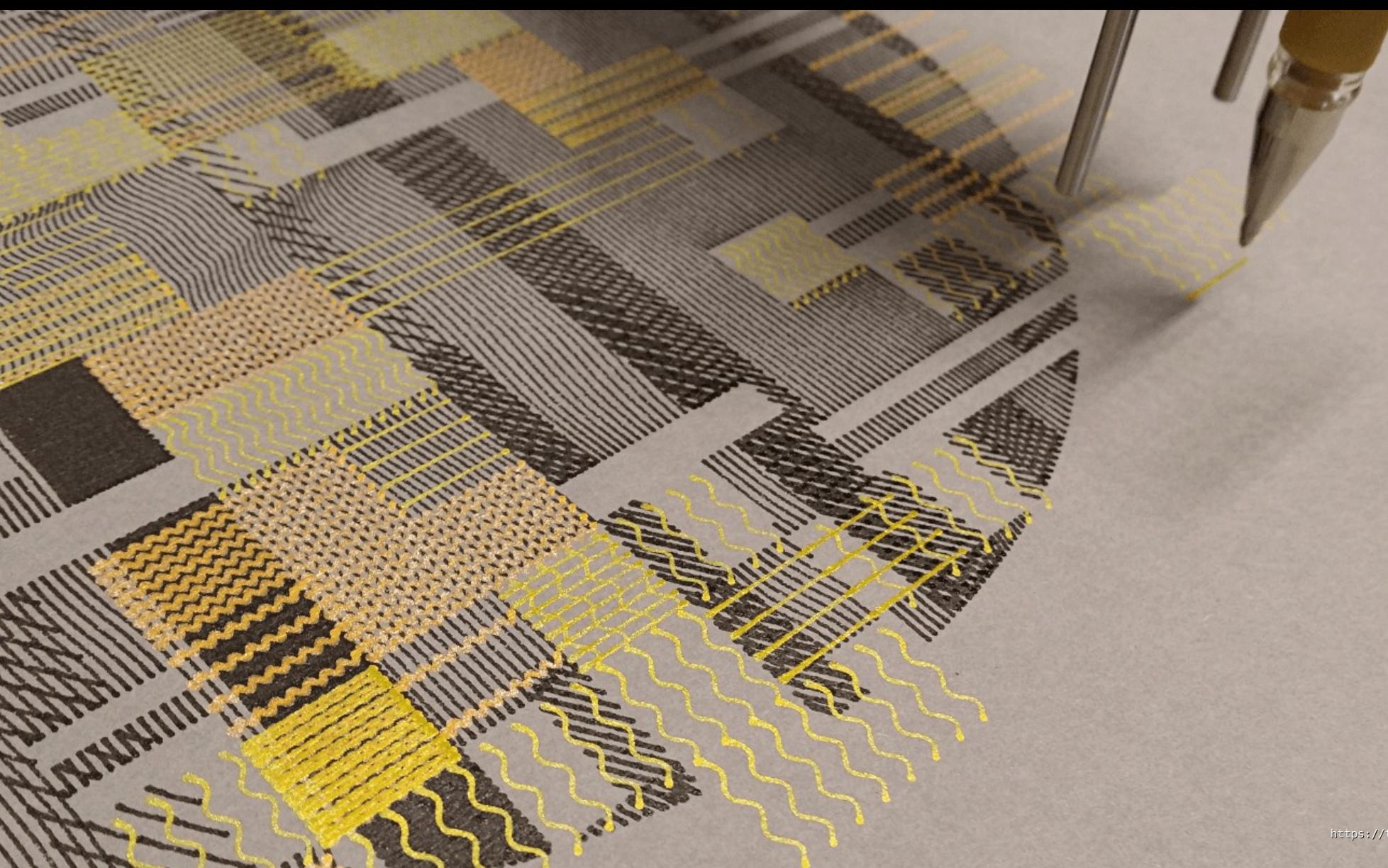
<https://create.arduino.cc/projecthub/>

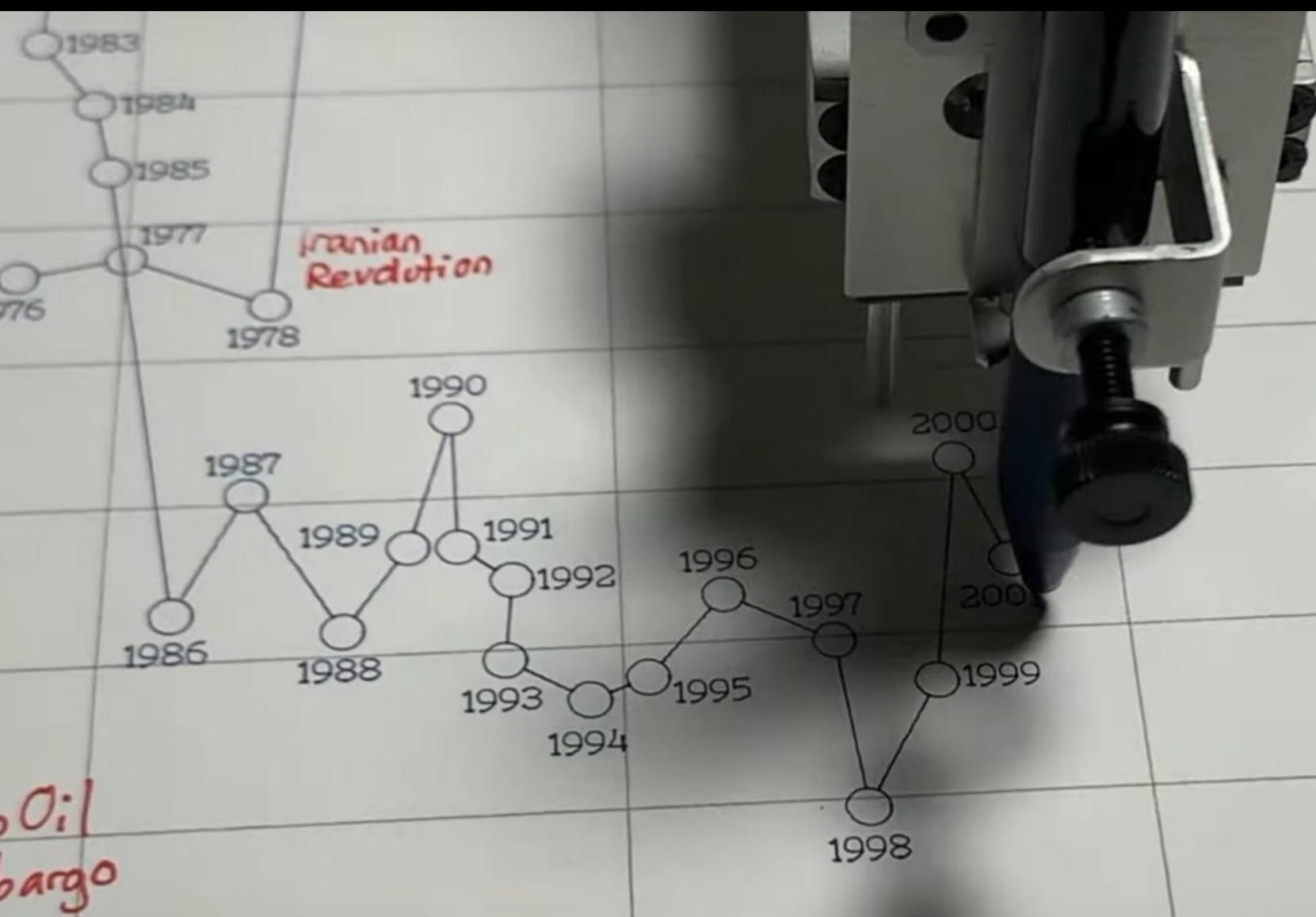
./processing_work/plotter_art





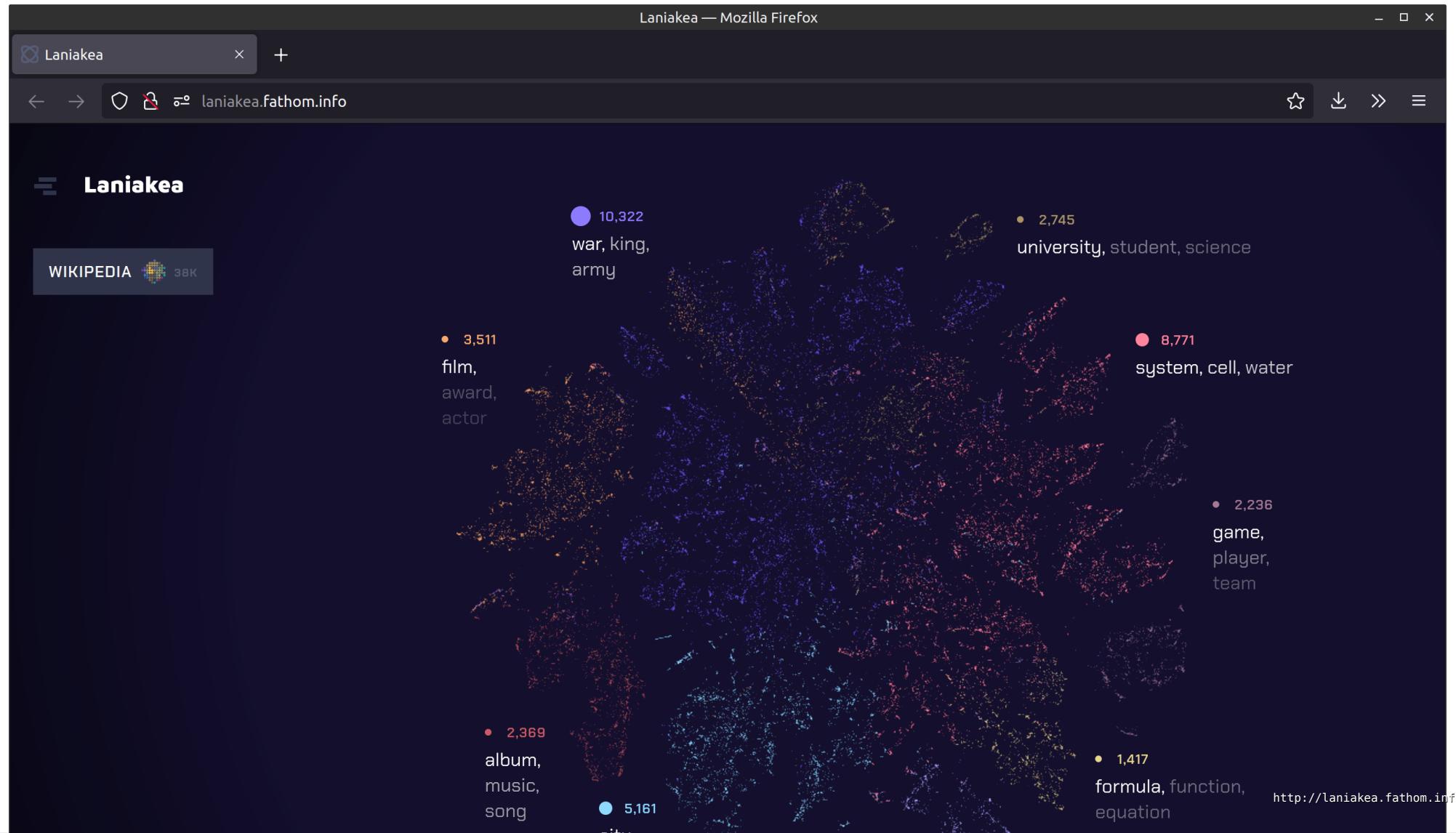
./processing_work/plotter_art

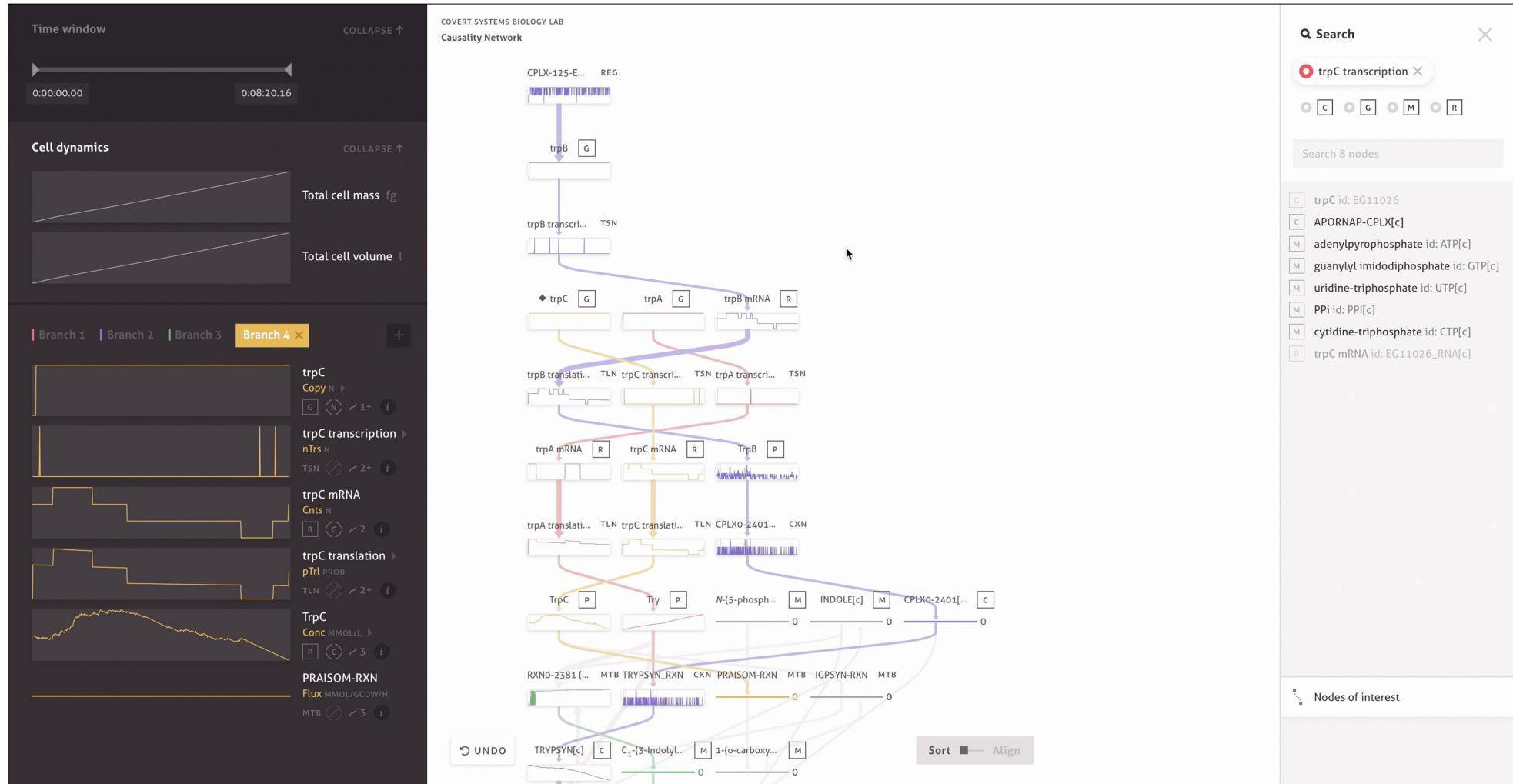


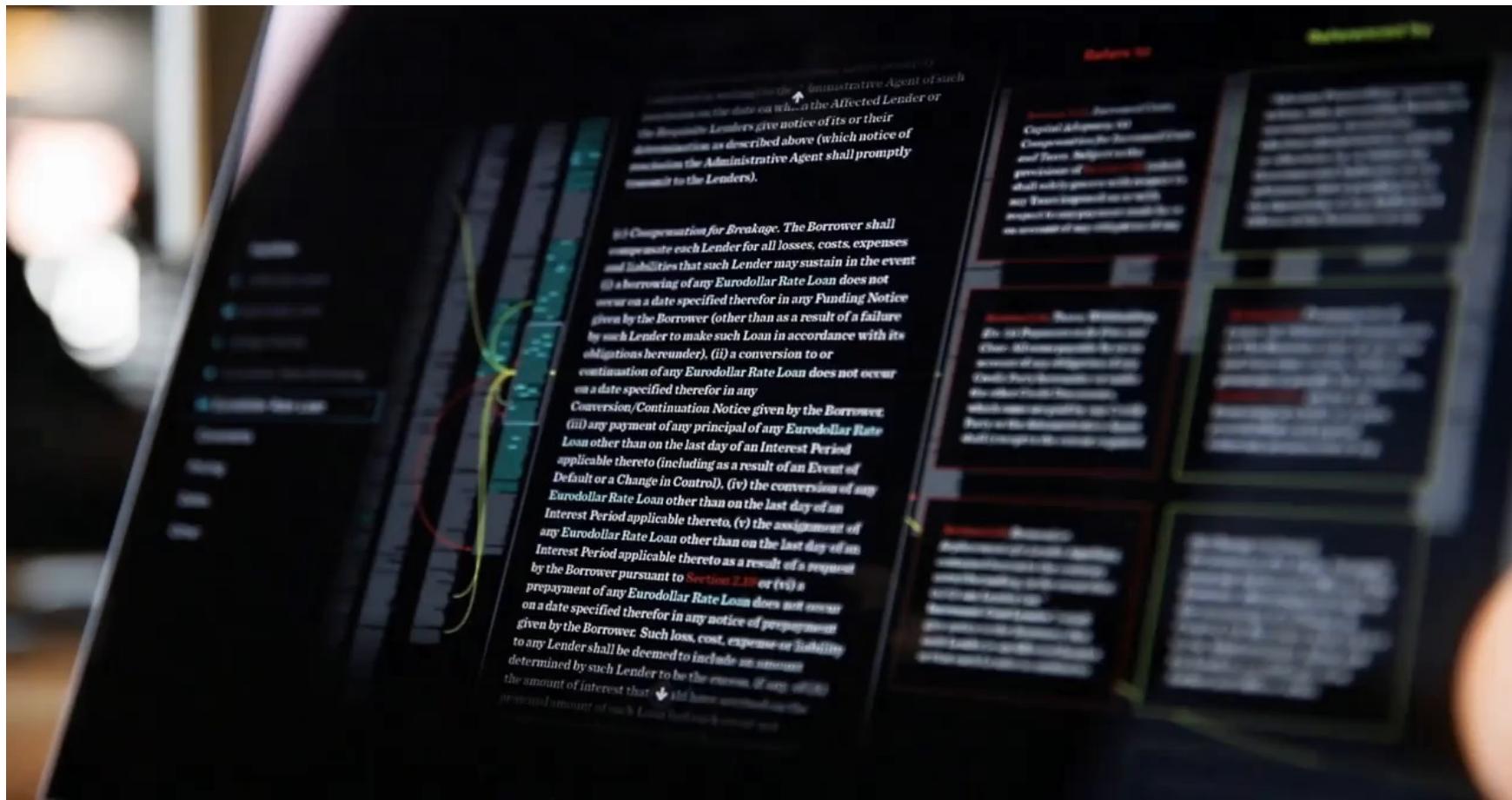


./processing_work_dataviz

fathom.info







./py5

py5

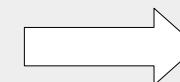
"py5 is a new version of processing for python 3.8+. it makes the java processing jars available to the cpython interpreter using jpyre. it can do just about everything processing can do, except with python instead of java code."

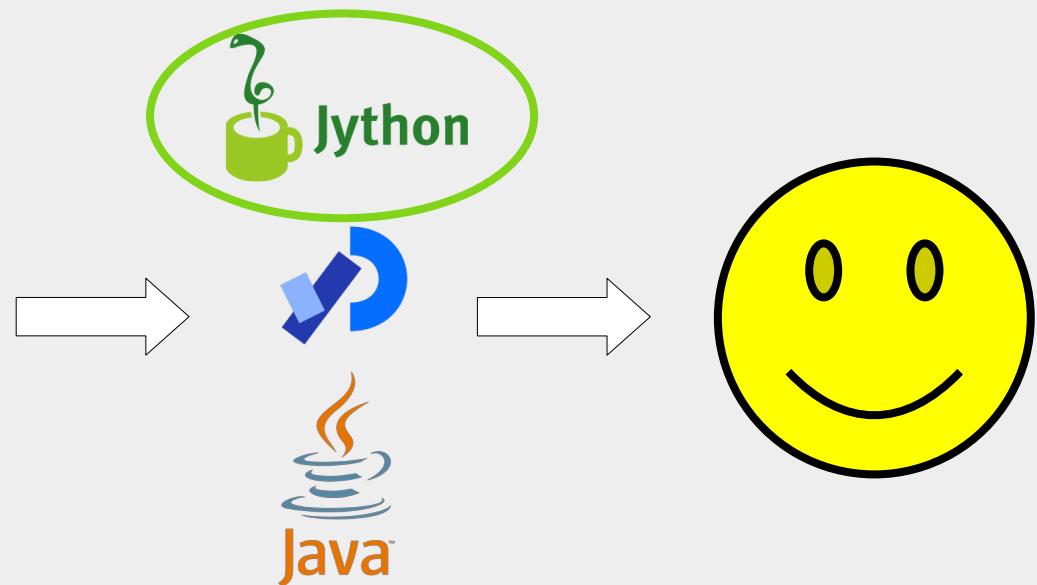


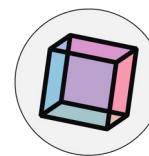
A screenshot of the Processing IDE interface. The title bar reads "sketch_220123d | Processing 4.0b3". The menu bar includes "File", "Edit", "Sketch", "Debug", "Tools", and "Help". Below the menu is a toolbar with play/pause buttons and a "Java" dropdown. The main workspace shows a code editor with the following text:

```
1 processing java code  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17
```

The bottom of the window features tabs for "Console" and "Errors".



A screenshot of the Processing Python IDE. The main window shows a sketch titled "sketch_220123c" with the code "processing python code" written in it. The code editor has line numbers from 1 to 17. Below the code editor is a dark grey workspace. At the bottom of the interface is a blue toolbar with a play button, a stop button, and a "Console" tab.



py5

sketch_210819a.py - VSCode

File Edit Selection View Go Run Terminal Help

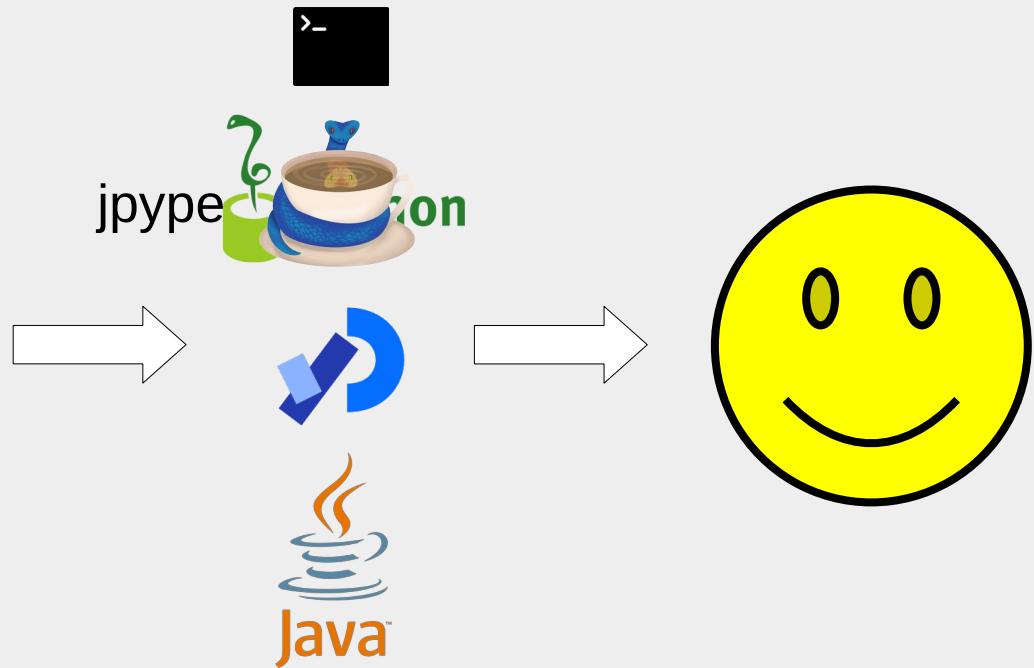
sketch_210819a.py ×

home > nuc > Desktop > sketch_210819a.py

1 |

python code

Ln 1, Col 1 Spaces: 4 UTF-8 LF Python ⚙



./py5/thonny_plug-in

The screenshot shows the Thonny Python IDE interface. The main window title is "Thonny - /home/nuc/Desktop/demo.py @ 6:35". The menu bar includes File, Edit, View, Run, Tools, Help, and py5. Below the menu is a toolbar with icons for file operations, run, and stop. An open editor tab is titled "demo.py" and contains the following code:

```
1 def setup():
2     size(300, 200)
3     rect_mode(CENTER)
4
5 def draw():
6     rect(mouse_x, mouse_y, 10, 10)
```

Below the editor is a "Shell" tab showing the command-line history:

```
Shell <Python 3.9.5 (bundled)>
>>> %cd /home/nuc/Desktop
>>> %Run /home/nuc/.local/lib/python3.9/site-packages/py5_tools/tools/run_sketch.py /home/nuc/Desktop/demo.py
>>> %cd /home/nuc/Desktop
>>> %Run /home/nuc/.local/lib/python3.9/site-packages/py5_tools/tools/run_sketch.py /home/nuc/Desktop/demo.py
```

A separate window titled "Sketch" is visible in the background, showing a blank white canvas with a small black square at the center, representing the current state of the Py5 sketch.



JupyterLab — Mozilla Firefox

File Edit View Run Kernel Tabs Settings Help

Introduction to py5bot.ipynb

py5bot

This is py5bot. A simple and easy to use programming environment for teaching the very basics of Python programming and creative coding with py5.

Each cell in this notebook can contain a series of py5 drawing commands. The drawing commands will be executed to create a static image that will be embedded in the notebook.

The main design goal is to provide a simple programming environment that is suitable for beginners. Individual programming concepts can be explained in isolation from more complicated Python concepts like functions or modules.

When hosted on Jupyter Hub and paired with Jupyter Lab's "Show Contextual Help" feature, py5bot can become an easy to use programming environment for educators to teach Python to beginners.

[launch binder](#)

Below is a simple example.

```
[1]: size(200, 200)
background(255, 255, 0)
rect(50, 50, 100, 100)
```

[1]:

A yellow square background with a white rectangle centered inside it, representing the output of the provided Py5 code.

./py5/info

py5 website:
py5.ixora.io

session resources:
github.com/tabreturn/py5-vuw-snap

./end

some examples