

creative coding with python and processing  
*//cc.fest.2021*

note to self: don't forget to start recording

creative coding with python and processing  
*//cc.fest.2021*

\* who has used (java) processing?

- \* who has used (java) processing?

- \* who has used p5.js?

- \* who has used (java) processing?
- \* who has used p5.js?
- \* who has used python?

setup

\* [github.com/tabreturn/processing.py-cc-fest](https://github.com/tabreturn/processing.py-cc-fest)



\* [with a python cc-test](#)

\* [tabreturn.com/cc](#)



Search or jump to...

Pull requests Issues Marketplace Explore



tabreturn / processing.py-cc-fest

Unwatch 1

Star 0

Fork 0

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags

Go to file

Add file

Code

tabreturn - add porting tips

0d190fd 1 minute ago 12 commits

README.md - add porting tips 1 minute ago

slides.pdf - add slides pdf 2 hours ago

README.md



# CC Fest 2021 Resources

Resources for my [Virtual CC Fest 2021 session](#) (Creative Coding with Python and Processing)

## Session activity

1 - Start here: <https://editor.computiful.org/tabreturn/sketches/V6gvX-LN8>

I'll be adding links, code, slides, and more soon ...

## About



Resources for my Virtual CC Fest session (Creative Coding with Python and Processing)

Readme

## Releases

No releases published  
[Create a new release](#)

## Packages

No packages published  
[Publish your first package](#)

processing 'ecosystem'

**Processing**

p5.js

Processing.py

Processing for Android

Processing for Pi

# Processing

Cover

Download

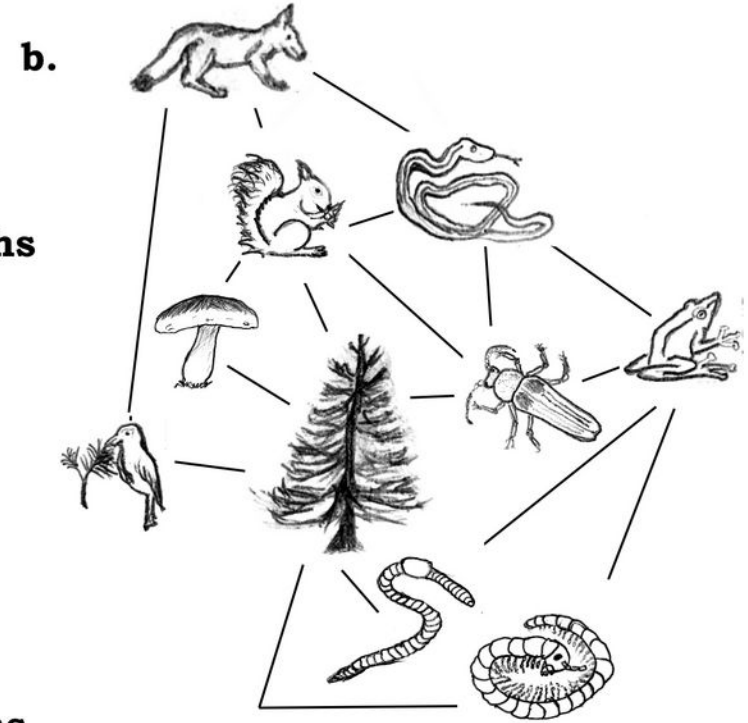
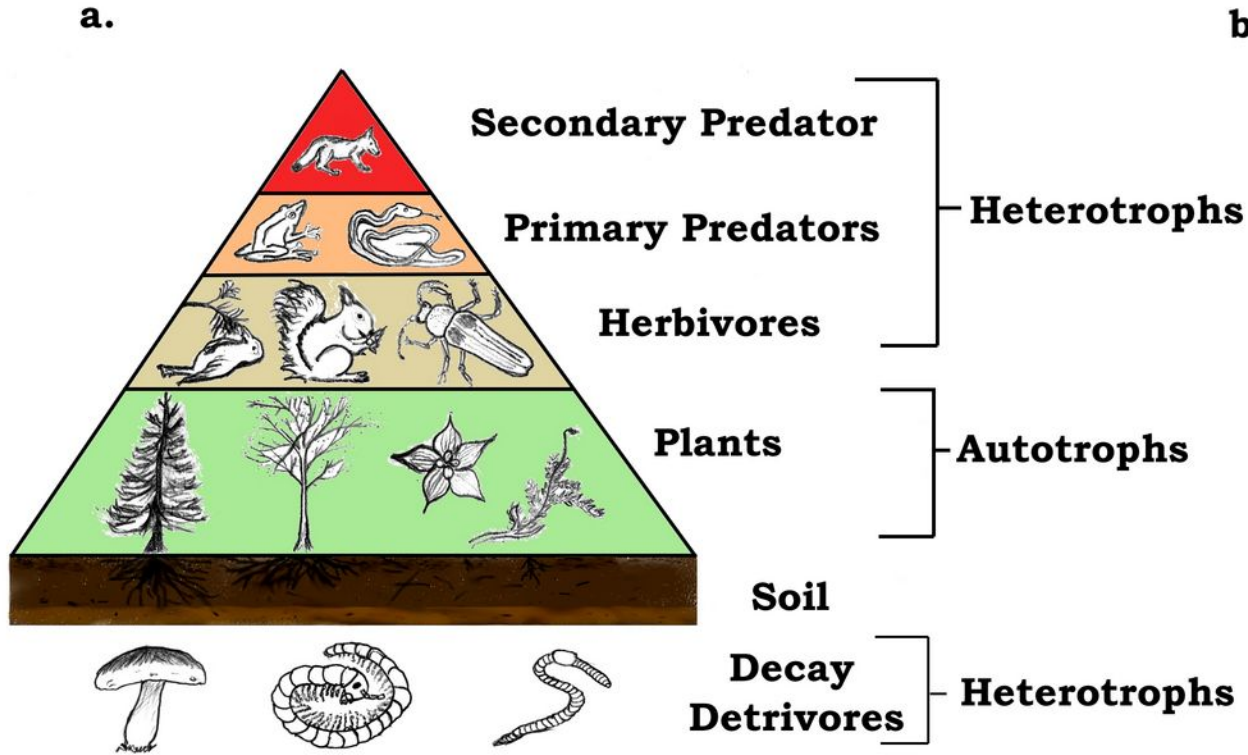
Donate

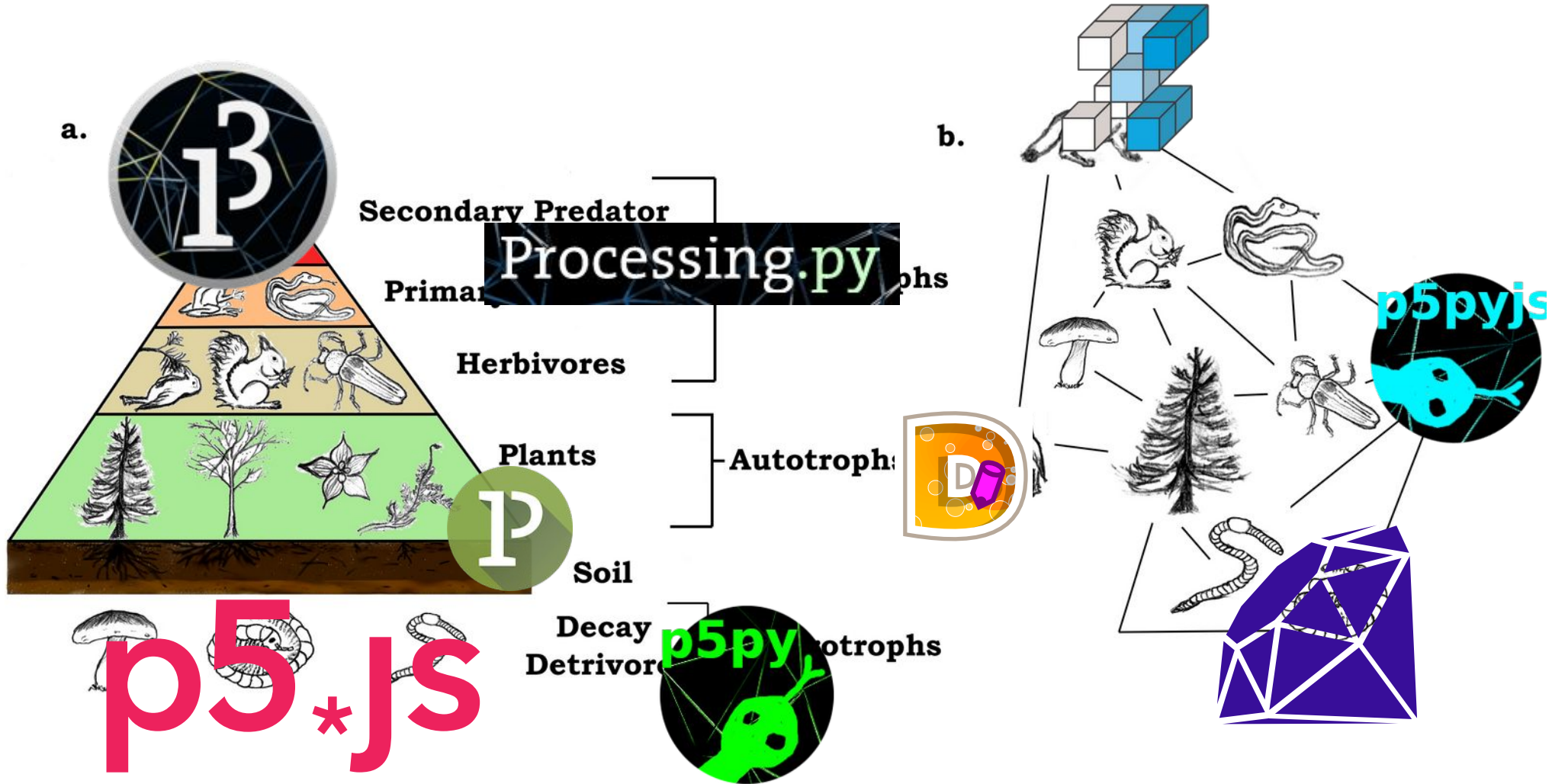
Reference

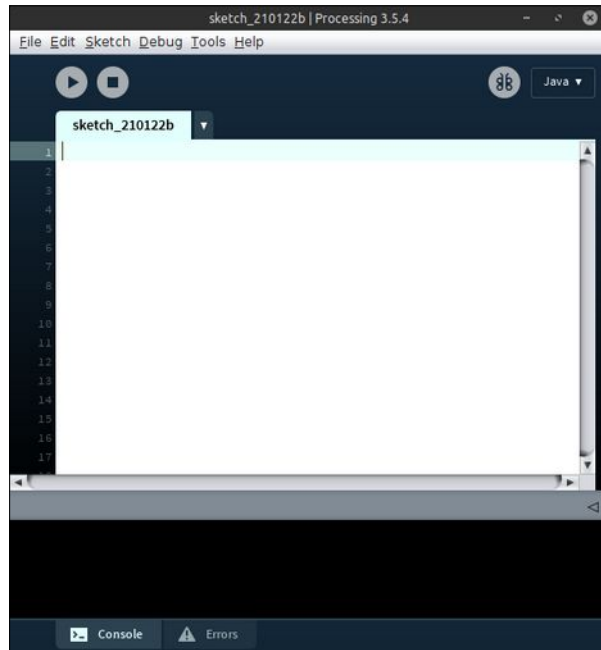
Libraries

Tools



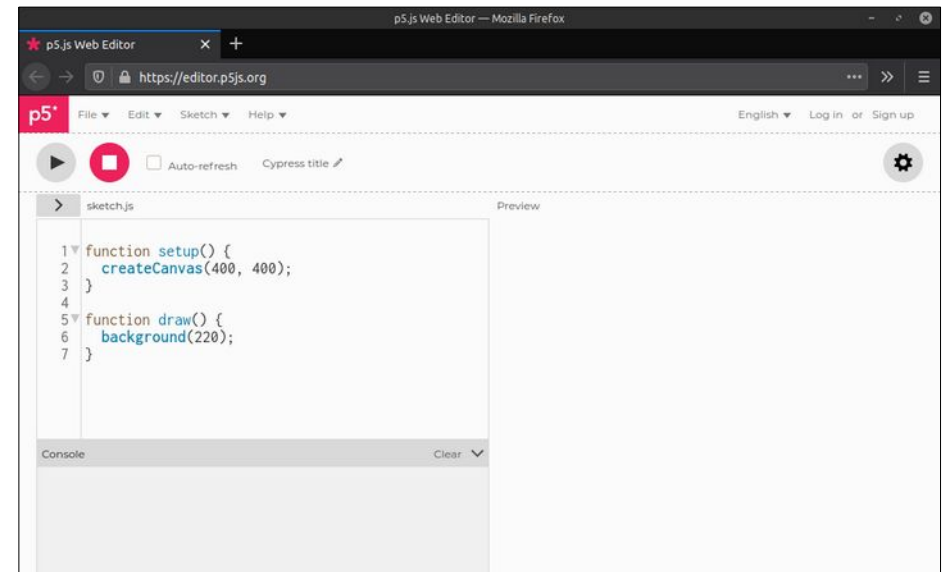
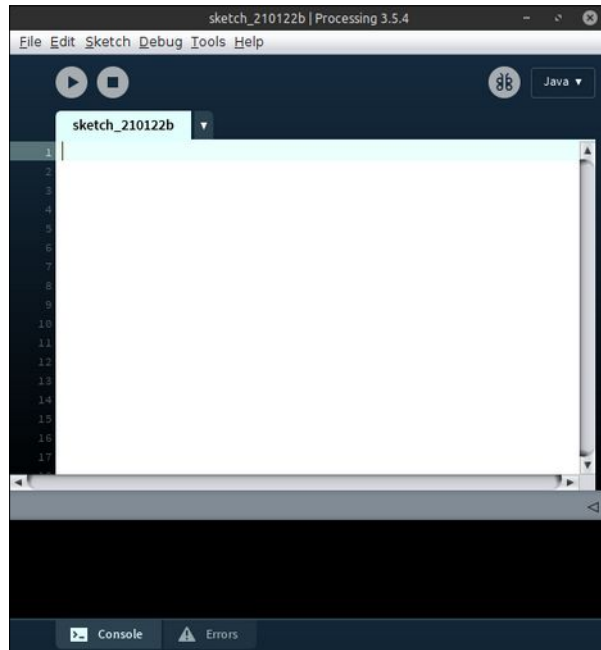




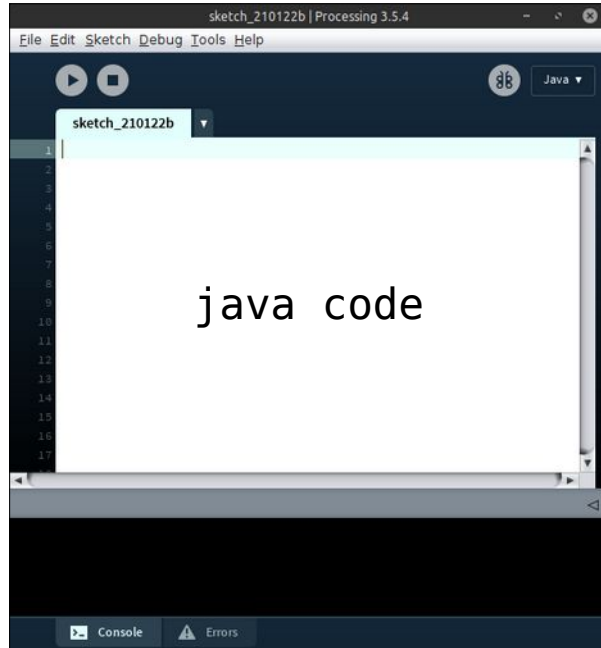




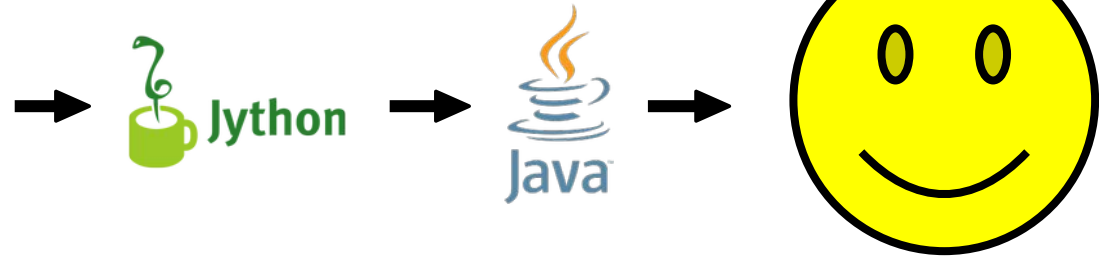
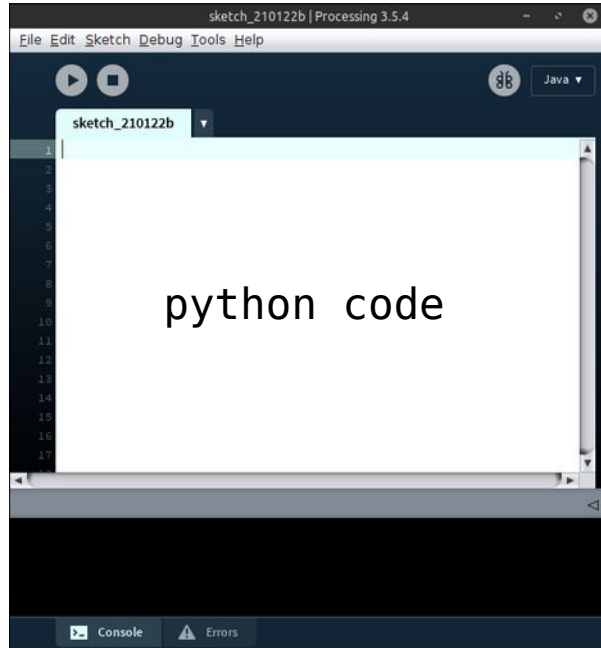
p5\*.Js







# Processing.py



[Processing](#)[p5.js](#)[Processing.py](#)[Processing for Android](#)[Processing for Pi](#)[Processing Foundation](#)

# Processing.py

[Cover](#)[Reference](#)[Tutorials](#)[Examples](#)[Bugs](#)

## Python Mode for Processing

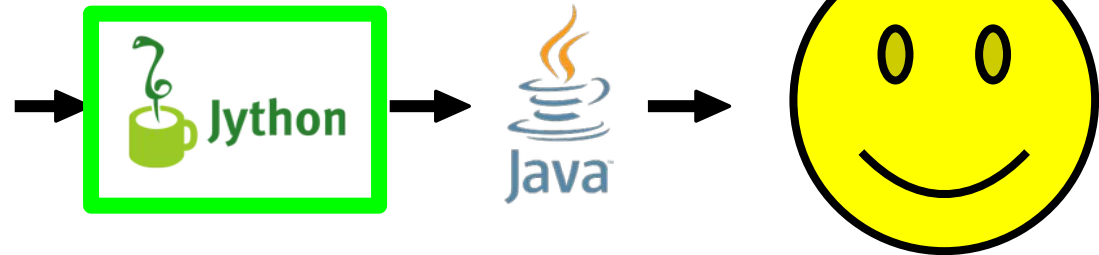
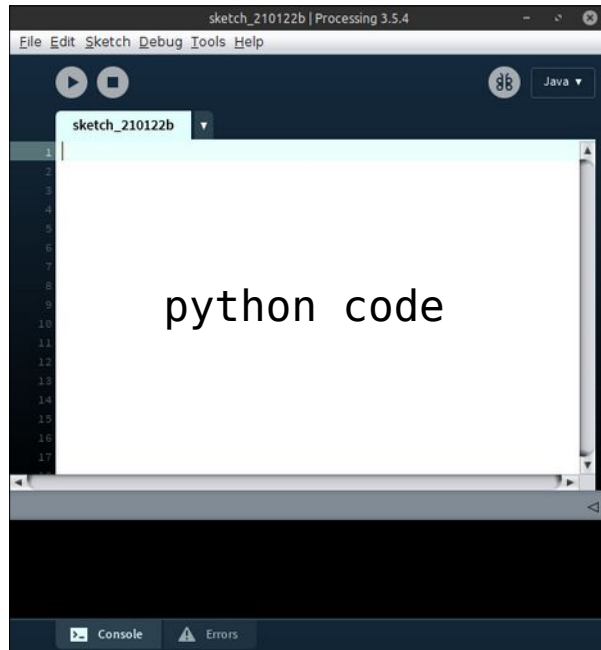
You write Processing code. In Python.

Processing is a programming language, development environment, and online community. Since 2001, Processing has promoted software literacy within the visual arts and visual literacy within technology. Today, there are tens of thousands of students, artists, designers, researchers, and hobbyists who use Processing for learning, prototyping, and production.

Processing was initially released with a Java-based syntax, and with a lexicon of graphical primitives that took inspiration from OpenGL, Postscript, Design by Numbers, and other sources. With the gradual addition of alternative programming interfaces — including [JavaScript](#), [Python](#), and [Ruby](#) — it has become increasingly clear that Processing is not a single language, but rather, an arts-oriented approach to learning, teaching, and making things with code.

We are thrilled to make available this public release of the Python Mode for Processing, and its associated documentation. More is on the way! If you'd like to help us improve the [implementation of Python Mode](#) and [its documentation](#), please find us on Github!

# Processing.py



- \* jython

- \* jython is not cpython (reference implementation)

- \* jython

- \* jython is not cpython (reference implementation)

- \* this has limitations, hence "python mode for processing"

- \* jython

- \* jython is not cpython (reference implementation)

- \* this has limitations, hence "python mode for processing"

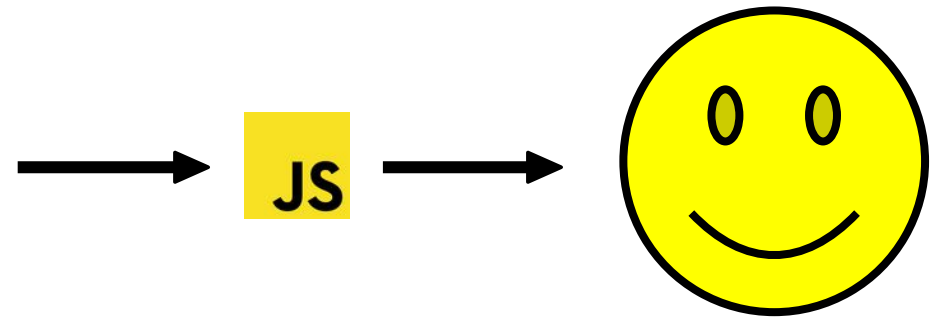
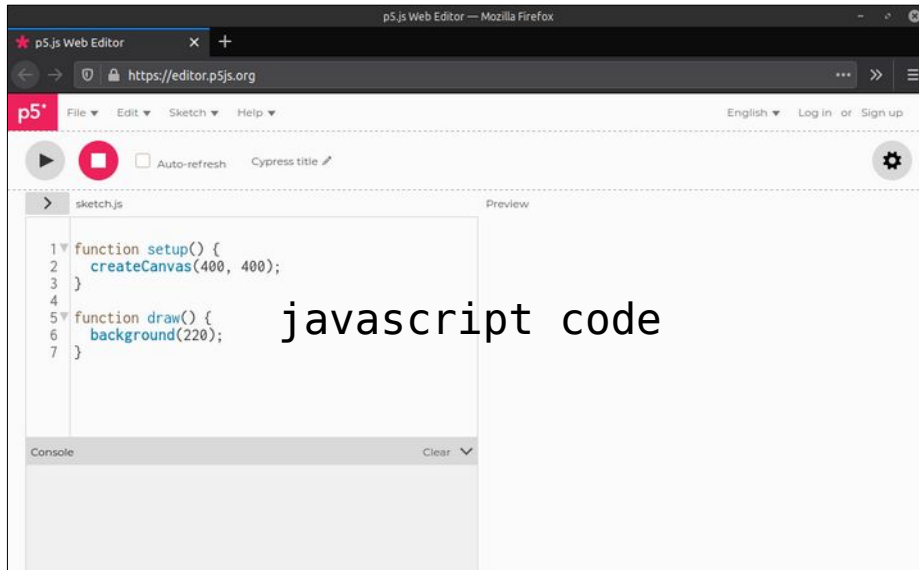
- \* more python implementations

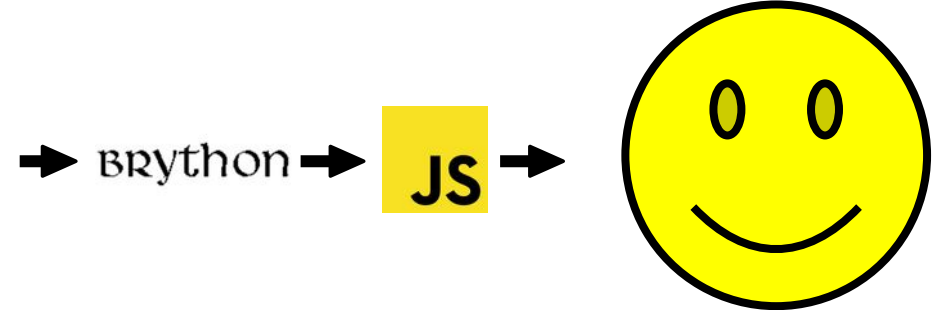
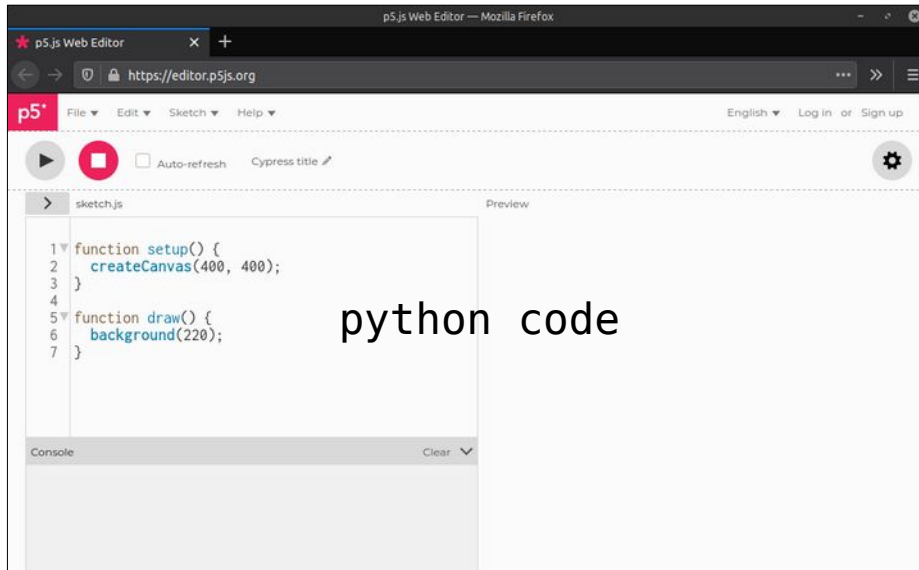
- \* micropython, ironpython, pypy, ...

- \* jython
  - \* jython is not cpython (reference implementation)
  - \* this has limitations, hence "python mode for processing"
- \* more python implementations
  - \* micropython, ironpython, pypy, ...
  - \* for javascript: transcript & brython



# p5\*.js





so:

- \* python mode / processing.py will do everything (java) processing does

SO:

- \* python mode / processing.py will do everything (java) processing does
- \* computiful will do everything p5.js does

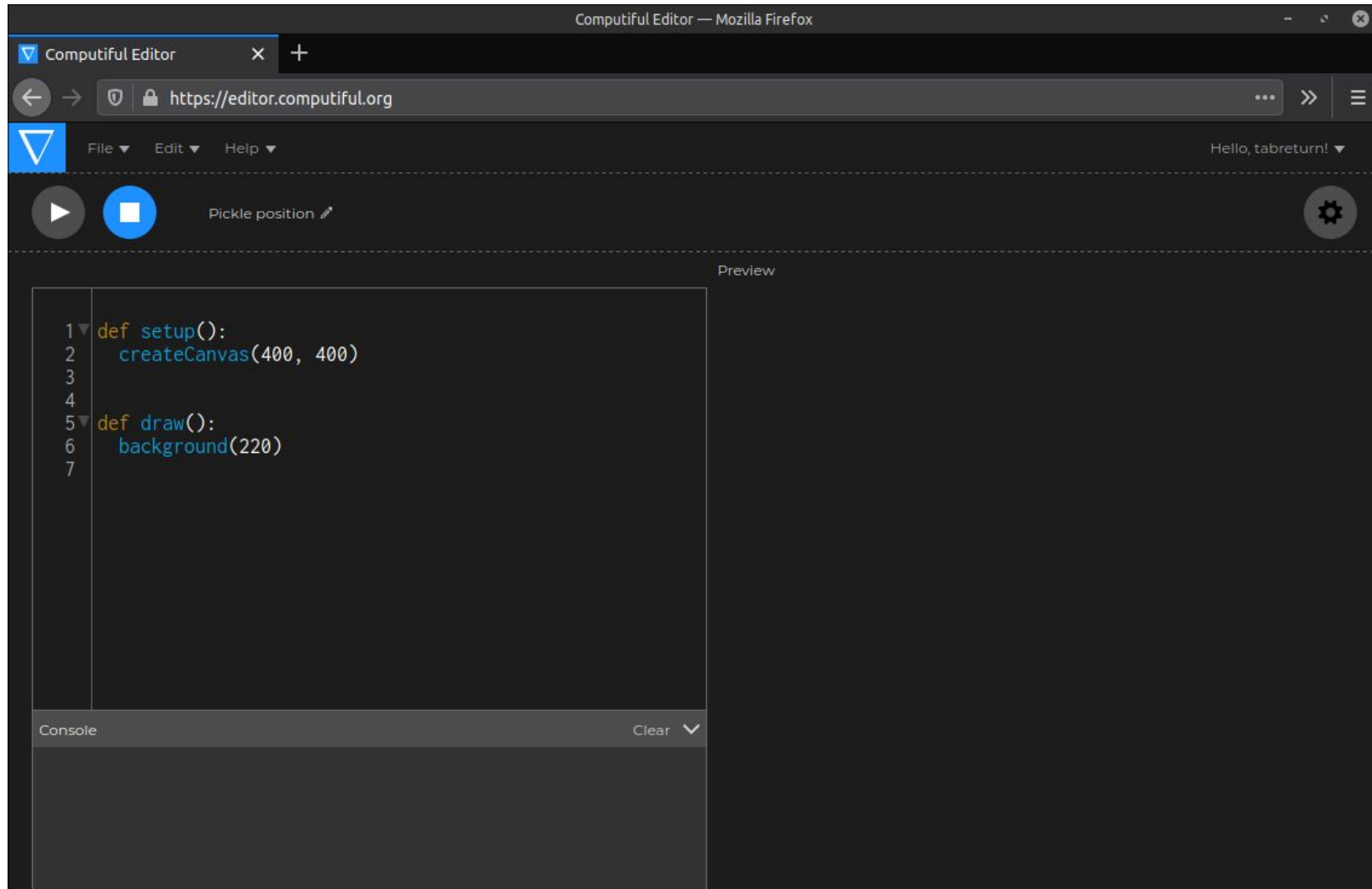
SO:

- \* python mode / processing.py will do everything (java) processing does
- \* computiful will do everything p5.js does
- \* today, we will use computiful

SO:

- \* python mode / processing.py will do everything (java) processing does
- \* computiful will do everything p5.js does
- \* today, we will use computiful

!!! a big thanks to nick mcintyre (who put computiful together) and the pyp5js crew (bernardo fontes, alexandre villares) and everybody else involved in those projects !!!

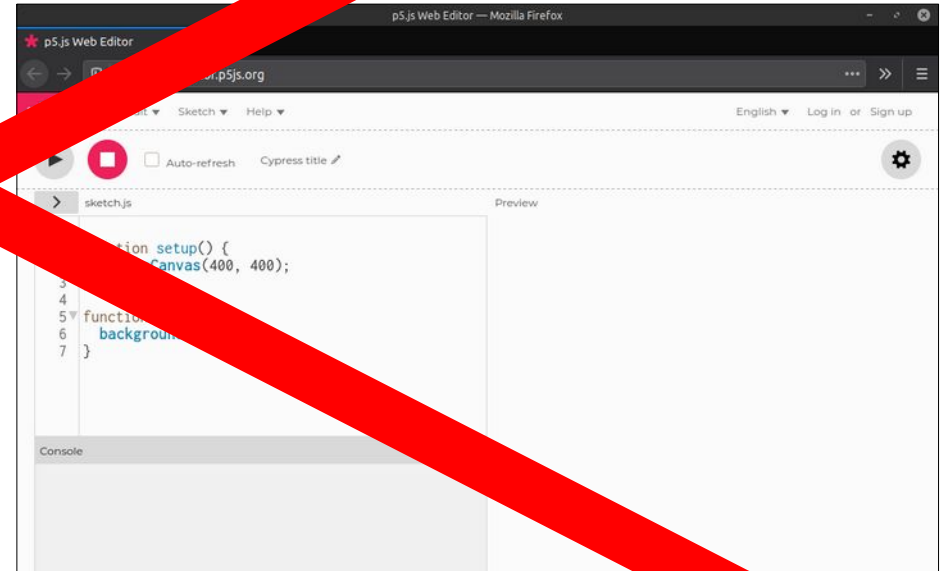
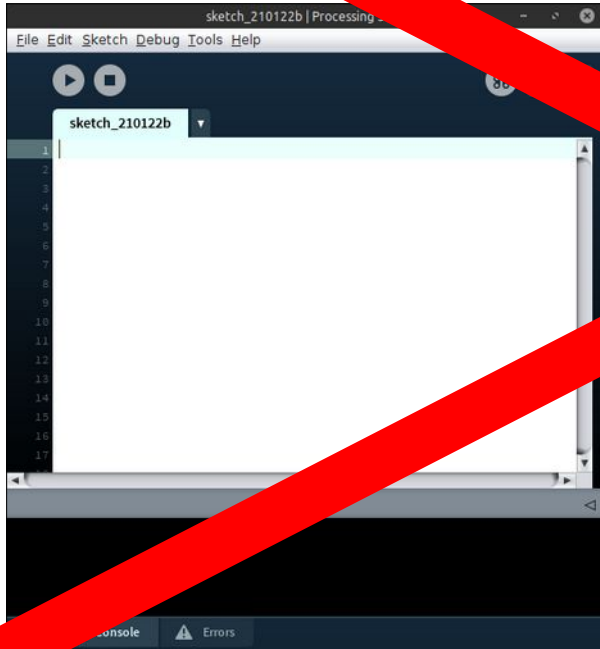


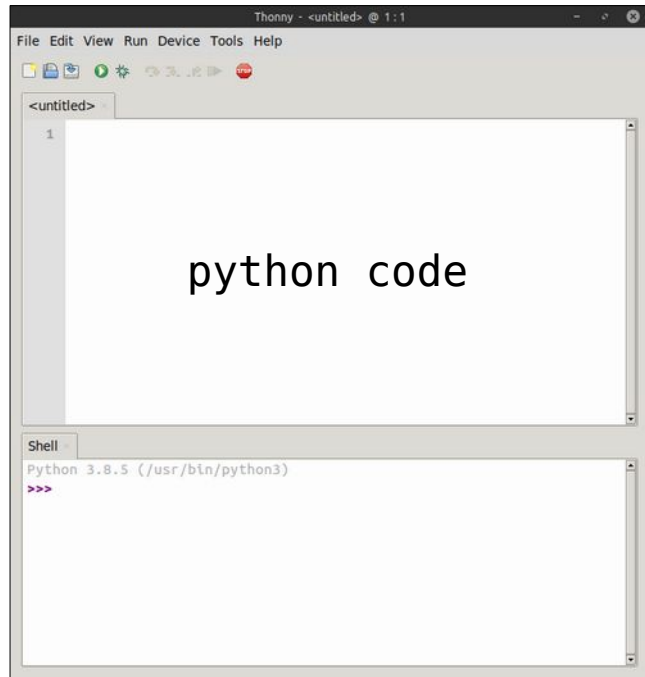
but, tristan -- what about a pure python solution?





p5\*.js





README.md	Update README.md	15 days ago
__config.yml	yaml	3 years ago

README.md

## Resources for teaching programming

### for artists, designers and architects ([repository](#))

#### Extended table of Hosts, Platforms & Languages

- Drawing or 3D modeling software that embeds a scripting language on the user interface or allows automation with a very limited number of steps between programming and code execution.
- Tools aimed at teaching programming in a visual or graphic context.
- [Ad hoc: tools that have shown potential for teaching in a visual or graphic context.]

<a href="#">SORTABLE VIEW OF THE TABLE</a>	<a href="#">CSV on GitHub</a>	<a href="#">raw CSV</a>
--	-------------------------------	-------------------------

An earlier version of this table was published as: VILLARES, A. B. A., & MOREIRA, D. (2017). *Python on the Landscape of Programming Tools for Design and Architectural Education*. Presented at the SIGRADI 2017, Concepcion, Chile.

Processing + Python tools table

Name	Processing features	based on (& Python version)	Python standard library	libraries ecosystem	main features	main limitations
<a href="#">Processing Python Mode</a>	<a href="#">Processing Java</a>	<a href="#">Jython</a> (Python 2)	complete	Java & Processing Java	available inside Processing IDE, very Processing compatible	no web deployment, no modern Python libs
<a href="#">p5py</a>	a new implementation (incomplete)	Python 3	complete	Python only	truly Python compatible	no web deployment, still incomplete
<a href="#">pyp5js</a> (transcrypt mode)	<a href="#">p5.js</a>	<a href="#">Transcrypt</a> (Python 3)	incomplete	JavaScript & p5.js	web ready sketches, very p5js compatible and nice browser editor	JS libraries only, p5.js features only (compared to Processing Java/Python modes)
					web ready	

Sponsor this project

<https://gumroad.com/villares>

Packages

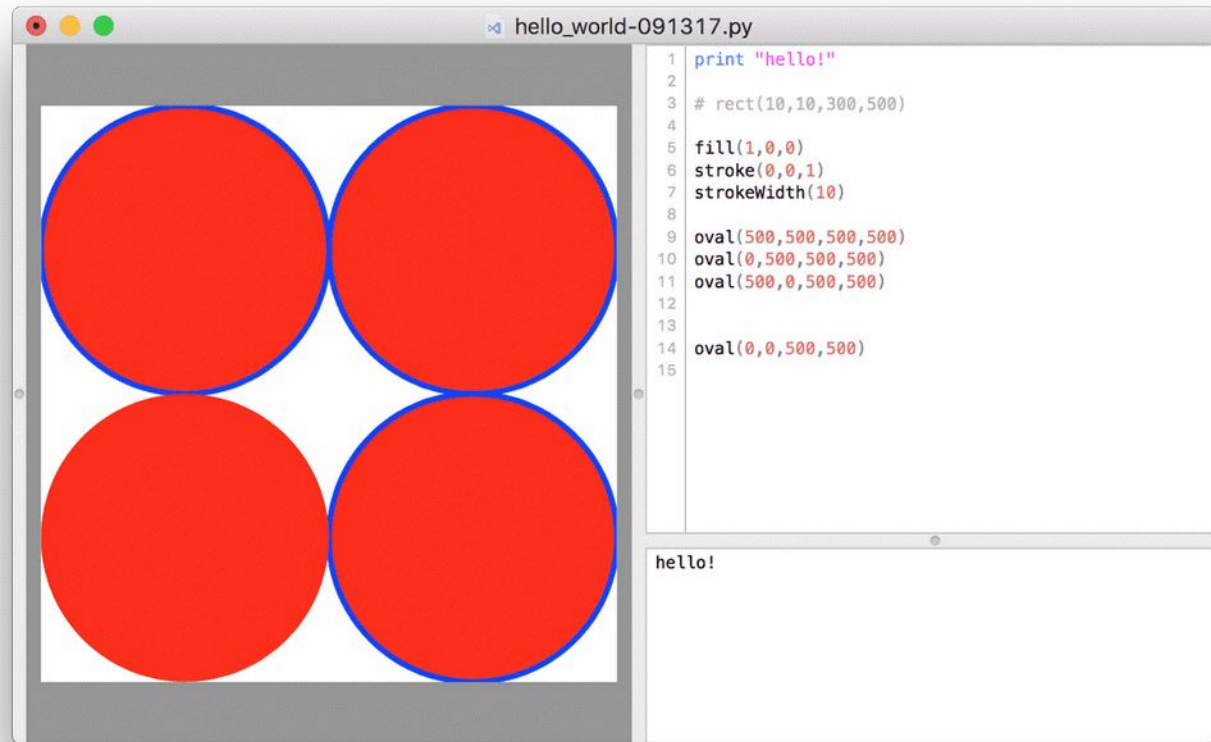
No packages published

Contributors 2

-  **villares** Alexandre B A Villares
-  **berinhard** Bernardo Fontes

Languages





forum:

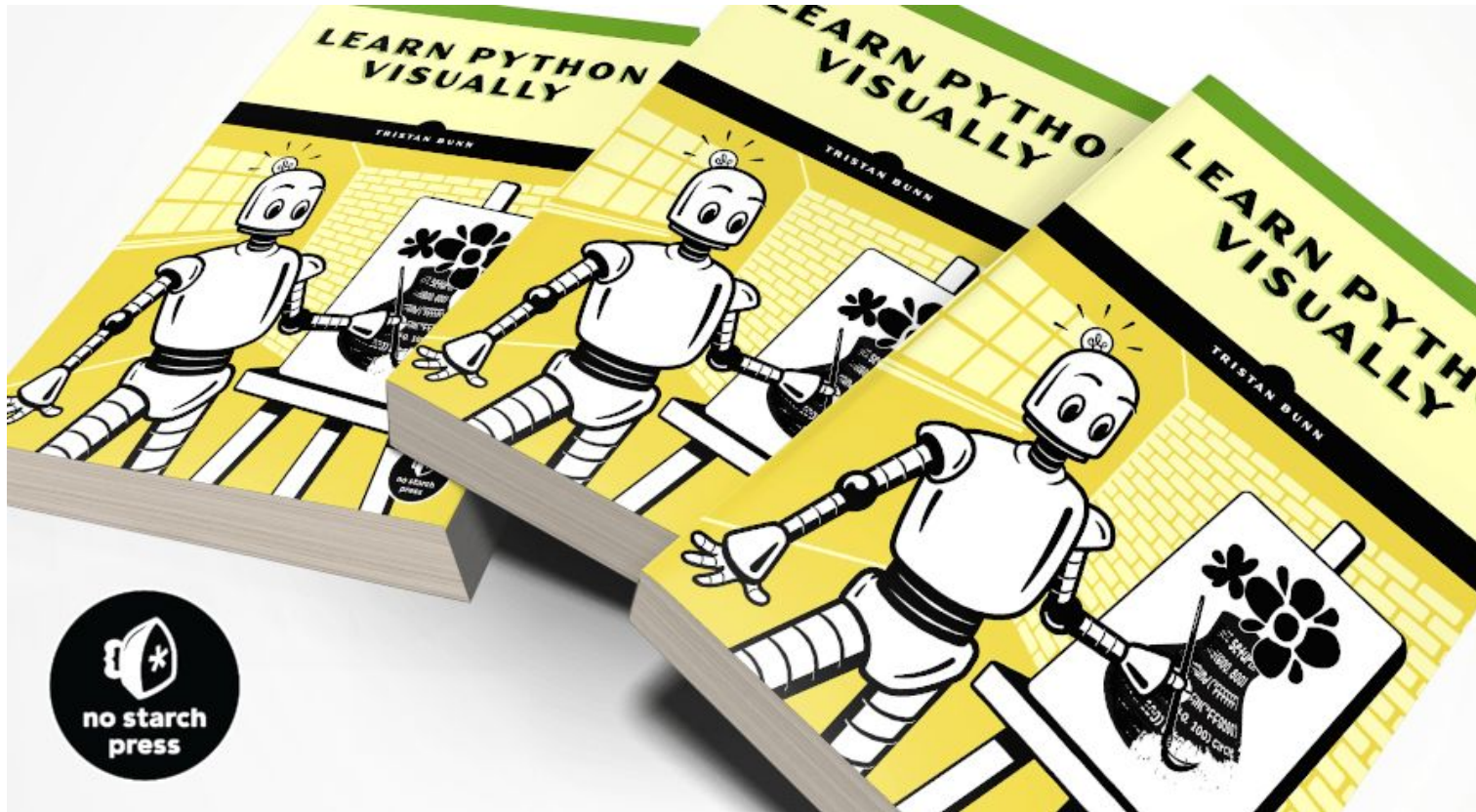
<https://discourse.processing.org/>

forum:

<https://discourse.processing.org/>

tips for porting processing (java) code to python mode:

[https://abav.lugaralgum.com/py.processing-play/java\\_to\\_python](https://abav.lugaralgum.com/py.processing-play/java_to_python)



```
./time_for_some_code
```

```
time for some code ...
```