



SWCON104
Web & Python Programming

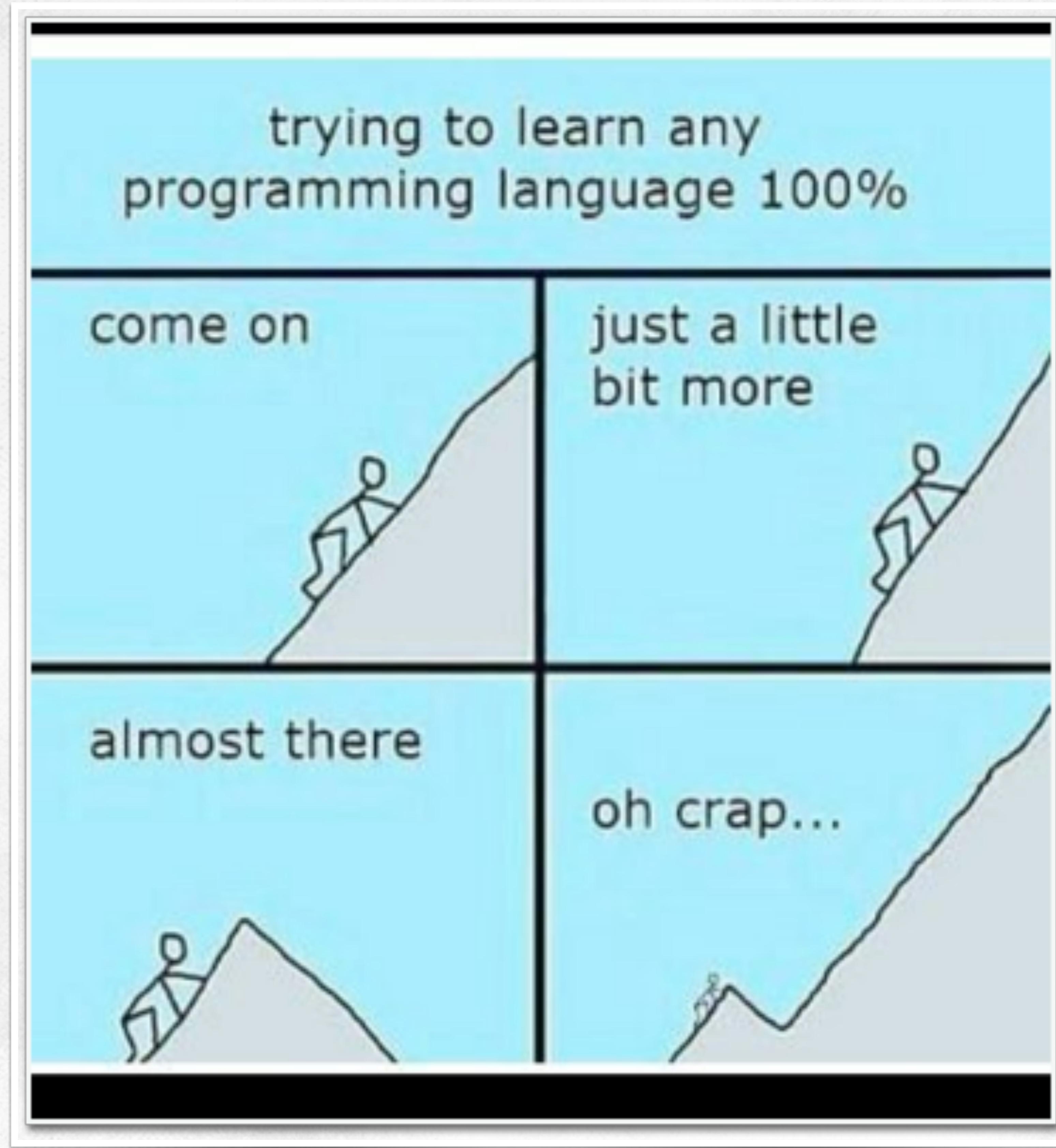
Wrapup

Department of Software Convergence

무엇을 배웠는가?

- 프로그래밍을 해야 하는 이유
- 오픈소스 소프트웨어 기반 개발 환경 구축
- Python을 통한 프로그래밍 언어의 이해
- Web 기술의 이해와 인터넷 통한 데이터 획득
- 데이터 분석 및 분석 결과의 시각화 도구 (3rd-party)
- 오픈소스 소프트웨어 라이브러리의 이해와 활용
- 제한된 시간안에서의 코딩 시험 경험
- 수업 게시판과 동영상을 통한 Self-learning 경험
- 프로그래밍 언어 Q&A를 위한 커뮤니티와 Googling 능력

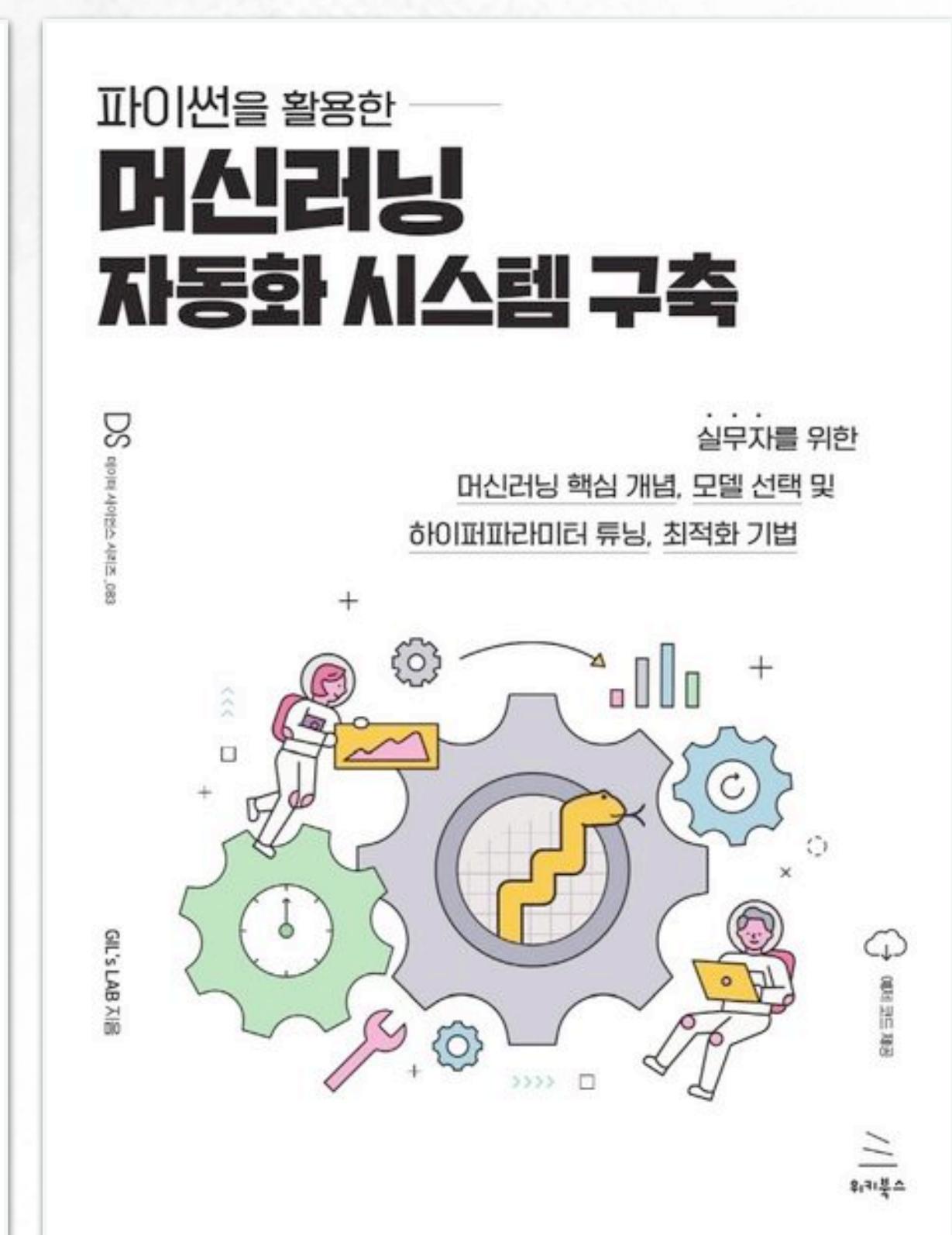
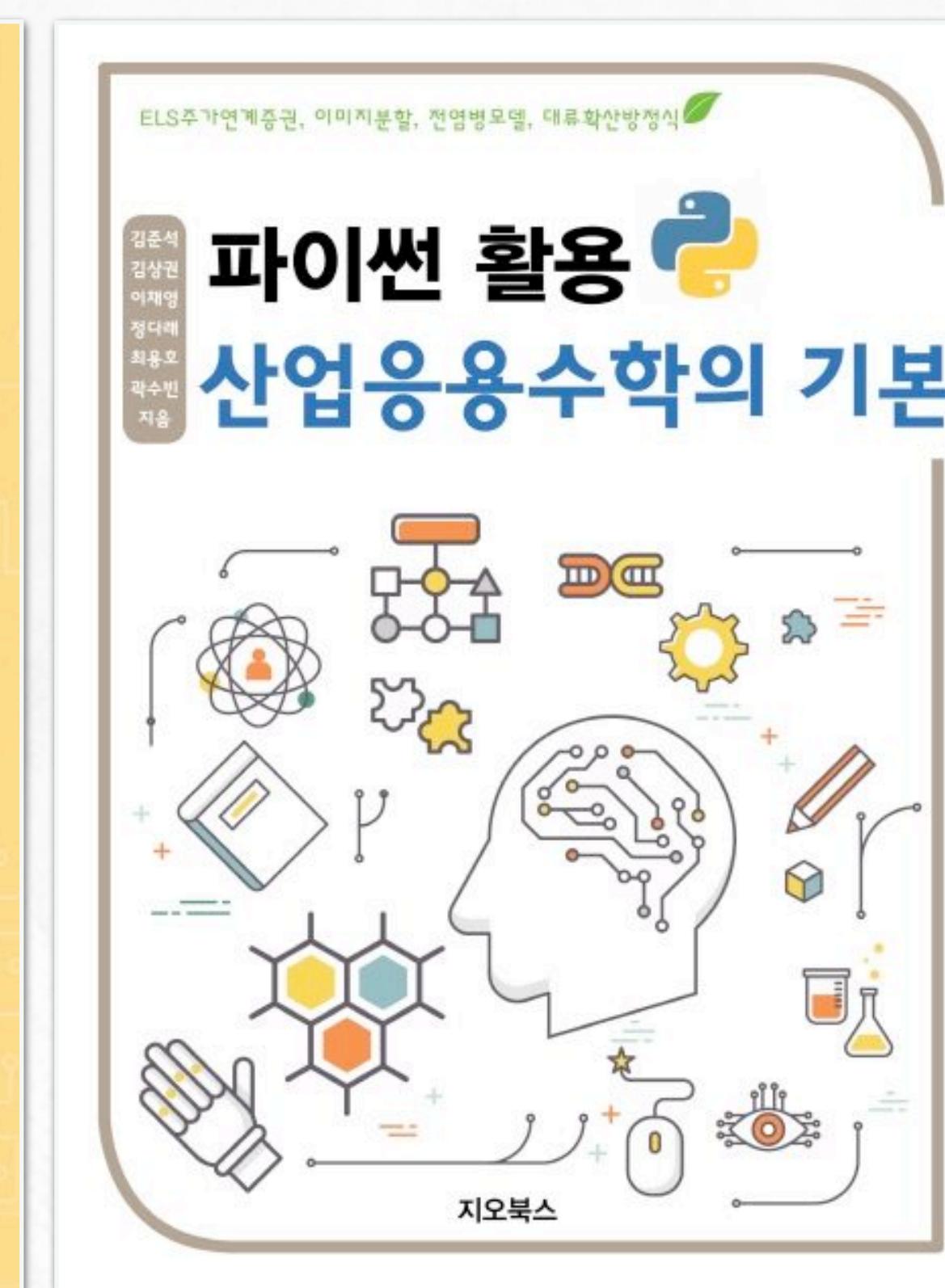
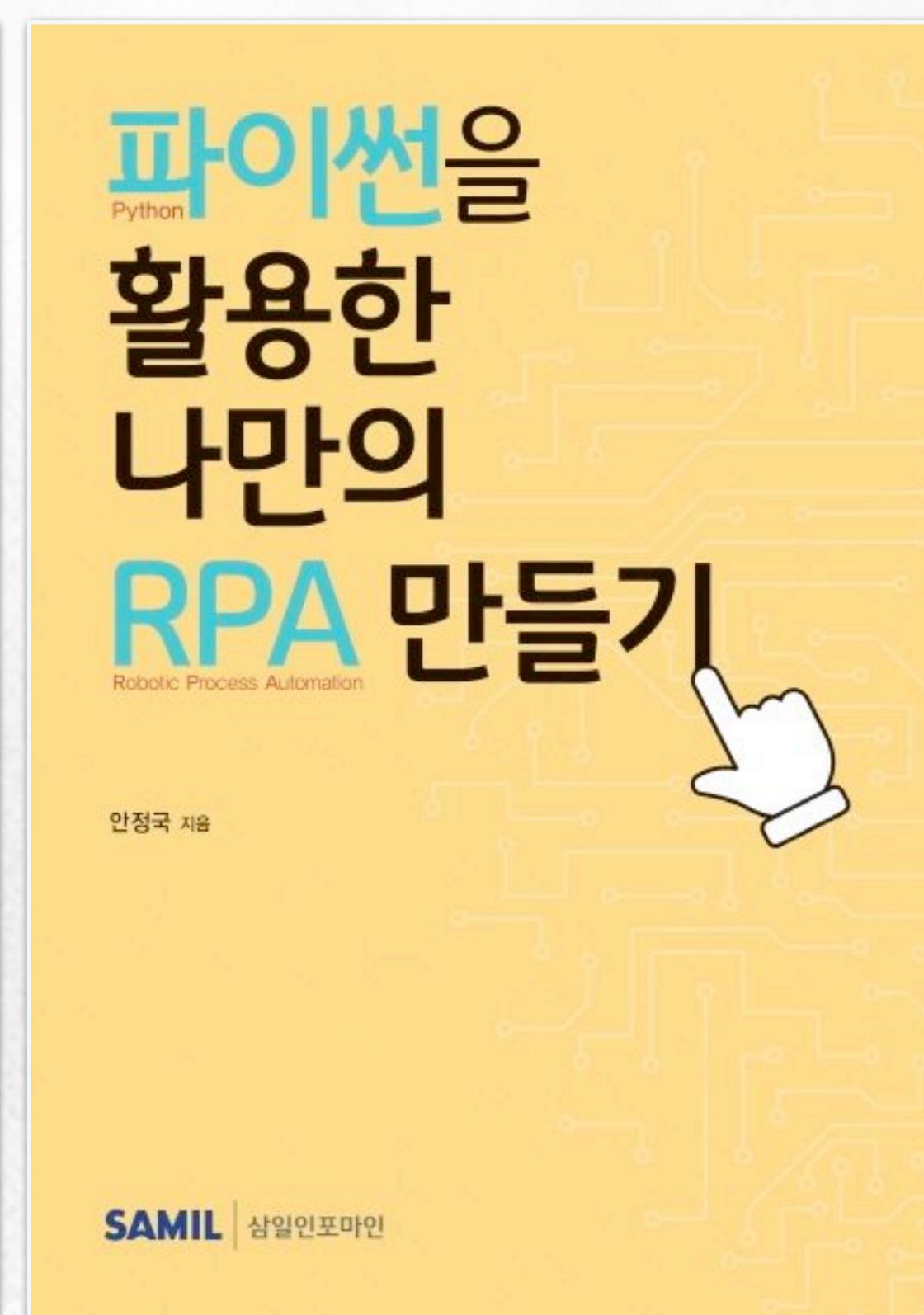
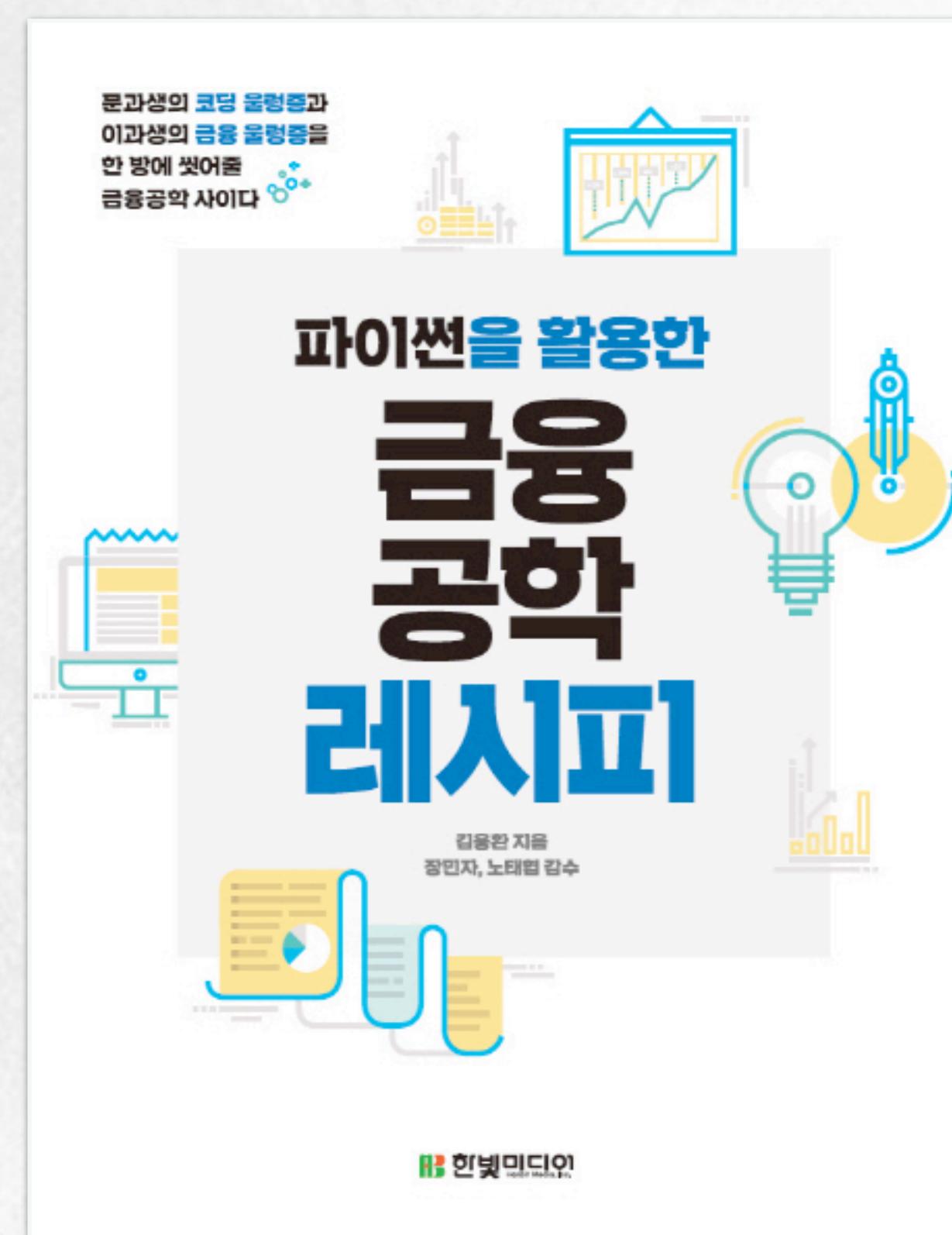
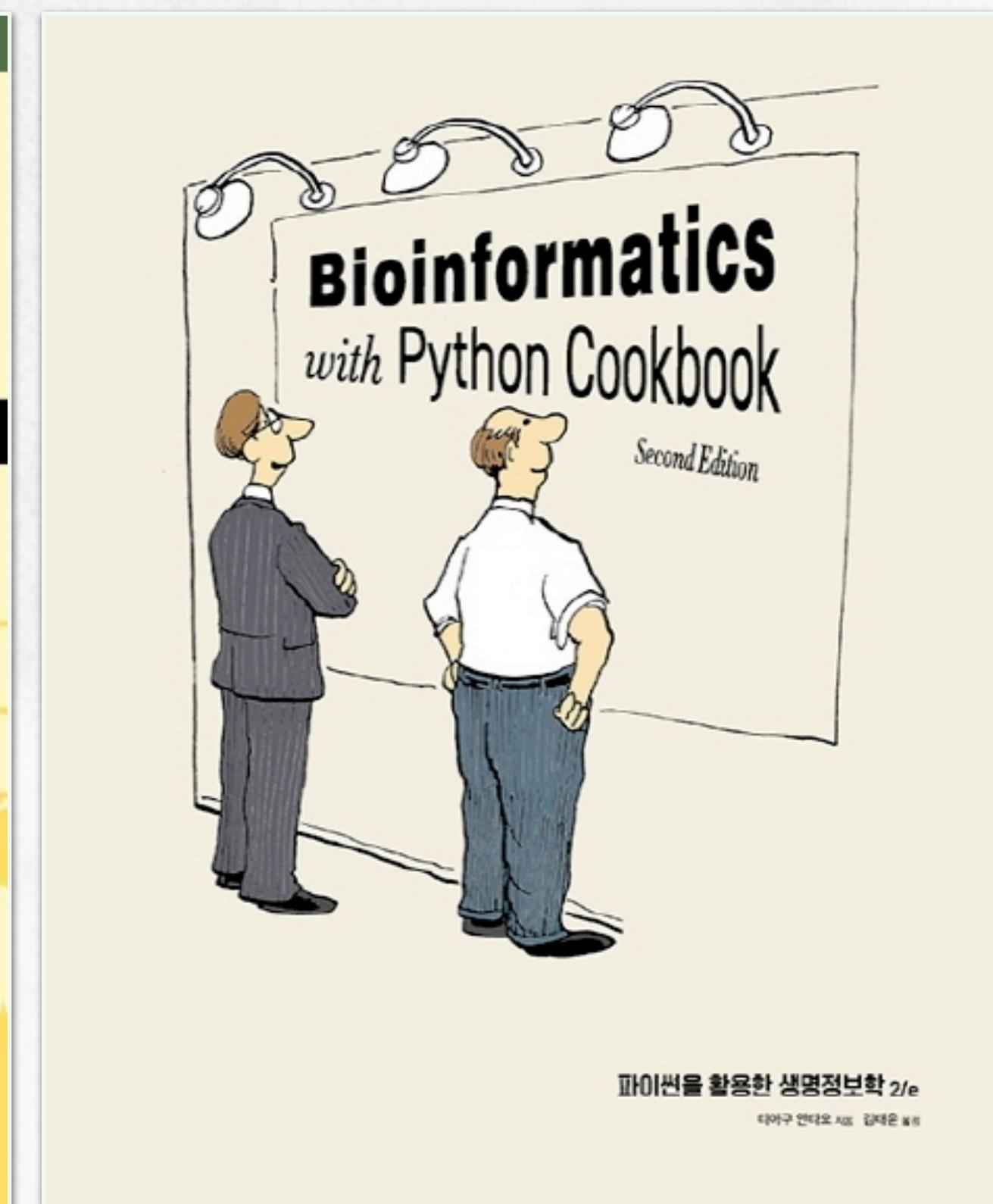
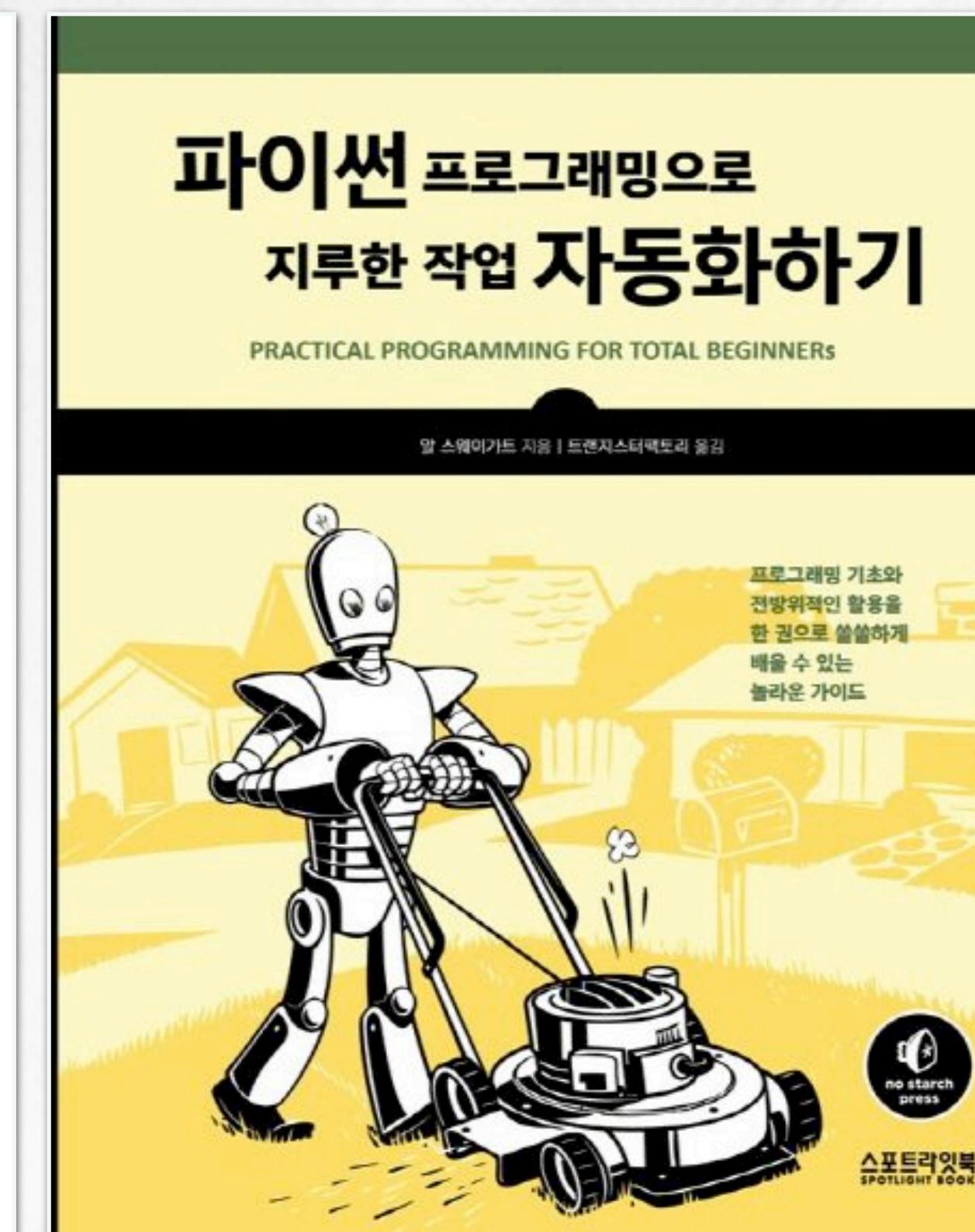
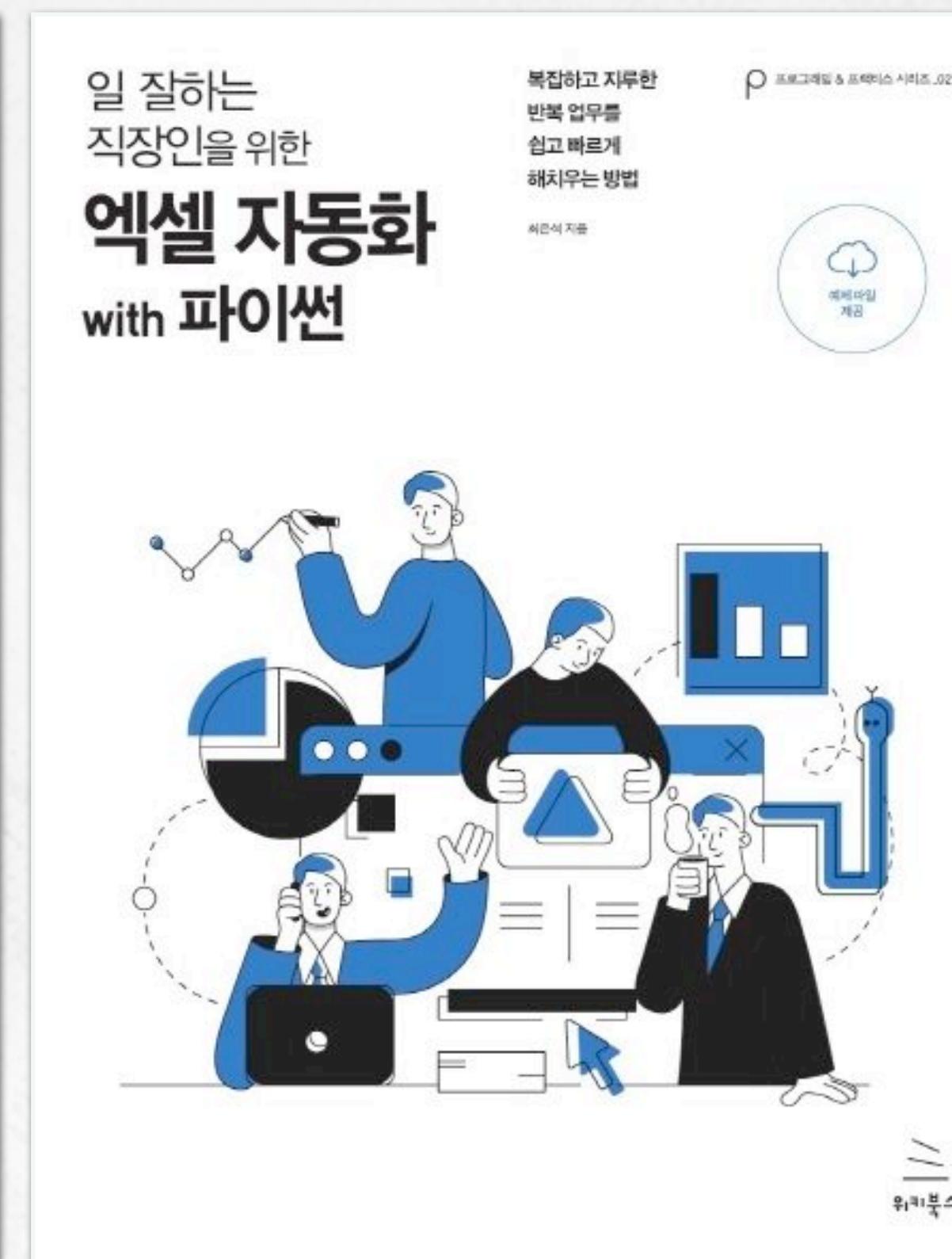
무엇이 남았는가?



무엇을 해야하나?

자기 주도 적인 문제정의
지속적인 자기 주도 학습
융합적인 팀 활동
의미있는 문제 해결과 결과물

실생활에서 Python 활용하기



지속 가능한 Python 개발자 되기

● Python.org 뉴스 레터 가입

The screenshot shows the Python Software Foundation (PSF) website. At the top, there is a navigation bar with links for Python, PSF, Docs, PyPI, Jobs, and Community. Below the navigation bar is the PSF logo and a search bar. A yellow call-to-action button says "Join the official Python Developers Survey 2022 and win valuable prizes: Start the Survey!". Below this, a breadcrumb navigation shows PSF >> Membership >> Newsletter. The main content area features a large heading "PSF Newsletter Signup". It explains that users can sign up for a bi-monthly newsletter to receive community news and provides a link to view past issues. A "Subscribe" form is present, asking for an "Email Address *". A note indicates that an asterisk (*) indicates required fields. On the left side, there is a sidebar titled "Tweets from @ThePSF" showing a tweet from the Python Software Foundation (@ThePSF) about the Python Developers survey.

Join the official Python Developers Survey 2022 and win valuable prizes: Start the Survey!

Python Developers Survey 2022

PSF >> Membership >> Newsletter

PSF Newsletter Signup

Sign up for our bi-monthly newsletter to receive important community news. You can view past issues [here](#). Follow us @ThePSF for that latest information.

Subscribe

* indicates required

Email Address *

Python Sof... 20h
@ThePSF

What's scarier than not having your voice heard? 😢 Make sure you let us know what you think in this year's annual Python Developers survey. 😁

surveys.jetbr...

지속 가능한 Python 개발자 되기

Python.org 뉴스 레터 가입 - 최신 버전 정보 상시 확인

Latest News

» More

Python Software Foundation - October 2022 Newsletter

Oct. 25, 2022

Python 3.12.0 alpha 1 released

Oct. 25, 2022

Python 3.11.0 is now available

Oct. 24, 2022

Python Core Development Sprint 2022: 3.11 and beyond!

Oct. 24, 2022

Python versions 3.10.8, 3.9.15, 3.8.15, 3.7.15 now available

Oct. 11, 2022

Monday, October 24, 2022
Python 3.11.0 is now available



This is the release of Python 3.11.0

Python 3.11 is finally released. In the CPython release team, we have put a lot of effort into making 3.11 the best version of Python possible. Better tracebacks, faster Python, exception groups and except*, typing improvements and much more. Get it here:

<https://www.python.org/downloads/release/python-3110/>

Major new features of the 3.11 series, compared to 3.10

Among the new major new features and changes so far:

- PEP 657 – Include Fine-Grained Error Locations in Tracebacks
- PEP 654 – Exception Groups and except*
- PEP 673 – Self Type
- PEP 646 – Variadic Generics
- PEP 680 – tomlib: Support for Parsing TOML in the Standard Library
- PEP 675 – Arbitrary Literal String Type
- PEP 655 – Marking individual TypedDict items as required or potentially-missing
- bpo-46752 – Introduce task groups to asyncio
- PEP 681 – Data Class Transforms
- bpo-433030 – Atomic grouping ((?>...)) and possessive quantifiers (*+, ++, ?+, {m,n}+) are now supported in regular expressions.
- The [Faster CPython Project](#) is already yielding some exciting results. Python 3.11 is up to 10-60% faster than Python 3.10. On average, we measured a 1.22x speedup on the standard benchmark suite. See [Faster CPython for details](#).

KYUNG HEE UNIVERSITY

Reference: <https://www.python.org/psf/newsletter/> 7

지속 가능한 Python 개발자 되기

- Google Alert에서 'Python' 콘텐츠 알리미 구독

The screenshot shows the Google Alert settings interface. At the top, there's a search bar with 'python'. Below it are several dropdown menus for filtering results:

- 수신 빈도: 하루에 한 번 이하
- 출처: 자동
- 언어: 모든 언어
- 지역: 모든 지역
- 개수: 가장 우수한 검색결과만
- 수신 위치: drsungwon@gmail.com

At the bottom of the settings panel are two buttons: '알림 만들기' (Create alert) and '옵션 숨기기 ▾' (Hide options).

Below the settings is a preview section titled '알리미 미리보기' (Alert preview). It shows a news item from NDTV.com about a Python吞食整只鹿的视频. The preview includes the title, source, a snippet of the article, and a small thumbnail image of a deer.

지속 가능한 Python 개발자 되기

● Real Python 사이트 가입

The screenshot shows the Real Python website. At the top, there's a dark header bar with the "Real Python" logo, a search bar, and "Join" and "Sign-In" buttons. Below the header, the main title "Real Python Tutorials" is displayed in large, bold letters. To the left of the main content, there's a large, colorful illustration of a woman interacting with a machine that has "doctest" written on it. The machine has sections labeled "DOCUMENTING...", "TESTING...", and "PASSED". The "TESTING..." section shows a green "OK" button being pressed. The "Real Python" logo is also present in the bottom right of the illustration. On the right side of the page, there's a sidebar with a "FREE Email Series" section titled "Python Tricks" featuring a snippet of Python code for merging dictionaries. Below that is a form for "Email...", a "Get Python Tricks »" button, and a note about no spam and unsubscribe options. At the bottom, there's a "All Tutorial Topics" section with various category buttons like "advanced", "api", "basics", etc.

Real Python Tutorials

— FREE Email Series —

Python Tricks ❤️

```
1 # How to merge two dicts
2 # in Python 3.5+
3
4>>> x = {'a': 1, 'b': 2}
5>>> y = {'b': 3, 'c': 4}
6
7>>> z = {**x, **y}
8
9>>> z
10{'c': 4, 'a': 1, 'b': 3}
```

Email...

Get Python Tricks »

No spam. Unsubscribe any time.

All Tutorial Topics

advanced api basics best-practices
community databases data-science
devops django docker flask front-end
gamedev gui intermediate
machine-learning projects python testing

Oct 31, 2022

Intermediate python testing

Improve Your Python

지속 가능한 Python 개발자 되기

● Medium 기사 사이트 가입 및 Python 관심 분야 등록

The screenshot shows the Medium homepage with a yellow header. The header features the Medium logo, navigation links for 'Our story', 'Membership', 'Write', 'Sign In', and a 'Get started' button. Below the header, a large, bold text 'Stay curious.' is displayed, followed by a subtext: 'Discover stories, thinking, and expertise from writers on any topic.' A 'Start reading' button is located at the bottom left of the yellow area. The main content area has a white background and displays several trending articles in a grid format. Each article includes a small profile picture, the author's name, the article title, and a brief description. The articles are numbered 01 through 06.

TRENDING ON MEDIUM

- 01** Dave Troy
No, Elon and Jack are not “competitors.” They’re collaborating.
Oct 29 · 9 min read
- 02** Tom Cooper
Ukraine War, 29 October, 2022: Q&A
Oct 29 · 11 min read
- 03** Simon Holdorf in Level Up Coding
10 Must-Read Books for Software Engineers
Oct 28 · 7 min read
- 04** Cory Doctorow
How to Leave Dying Social Media Platforms
- 05** Christianlauer in CodeX
Python 3.11.0 is released—Impacts to Data
- 06** Arthur Hayes in Entrepreneur's Handbook
Comeback

지속 가능한 Python 개발자 되기

● PyCon KOREA 참석하기



지속 가능한 Python 개발자 되기

관심 분야 YouTube 영상 시청

The screenshot shows the YouTube channel page for '라인개발실록' (Line Development Log). The channel has 1.22 million subscribers. The main navigation bar includes Home, Shorts, Subscriptions, and Watch later. The video feed displays four recent uploads:

- 최근에 업로드된 동영상** (Recent Uploads):
 - 라인개발실록 촬영 비하인드 썰 | 15:13 | 조회수 423회 · 12일 전
 - 테크니컬 라이터 전망이 어떤가요? | 라인 개발실록 | 25:32 | 조회수 390회 · 2주 전
 - 테크니컬 라이터에 관한 오해와 진실 | 라인 개발실록 | 24:42 | 조회수 388회 · 2주 전
 - 테크니컬 라이터가 하는 일 | 라인개발실록 | 20:49 | 조회수 821회 · 3주 전
- 인기 동영상** (Popular Videos): This section is currently empty.

Coding Guideline 익히기

- Google Python Coding Guideline
- <https://google.github.io/styleguide/pyguide.html>

styleguide

Google Python Style Guide

▶ Table of Contents

1 Background

Python is the main dynamic language used at Google. This style guide is a list of *dos and don'ts* for Python programs.

To help you format code correctly, we've created a [settings file for Vim](#). For Emacs, the default settings should be fine.

Many teams use the [yapf](#) auto-formatter to avoid arguing over formatting.

2 Python Language Rules

2.1 Lint

Run `pylint` over your code using this [pylintrc](#).

2.1.1 Definition

`pylint` is a tool for finding bugs and style problems in Python source code. It finds problems that are typically caught by a compiler for less dynamic languages like C and C++. Because of the dynamic nature of Python, some warnings may be incorrect; however, spurious warnings should be fairly infrequent.

2.1.2 Pros

Catches easy-to-miss errors like typos, using-vars-before-assignment, etc.

2.1.3 Cons

`pylint` isn't perfect. To take advantage of it, sometimes we'll need to write around it, suppress its warnings or fix it.

2.1.4 Decision

Make sure you run `pylint` on your code.

Coding Guideline 익히기

- Style Guide for Python Code [Python 공식 문서]
- <https://peps.python.org/pep-0008/>

PEP 8 – Style Guide for Python Code

Author: Guido van Rossum <guido at python.org>, Barry Warsaw <barry at python.org>, Nick Coghlan <ncoghlan at gmail.com>

Status: Active

Type: Process

Created: 05-Jul-2001

Post-History: 05-Jul-2001, 01-Aug-2013

▶ [Table of Contents](#)

Introduction

This document gives coding conventions for the Python code comprising the standard library in the main Python distribution. Please see the companion informational PEP describing [style guidelines for the C code in the C implementation of Python](#).

This document and [PEP 257](#) (Docstring Conventions) were adapted from Guido's original Python Style Guide essay, with some additions from Barry's style guide [2].

This style guide evolves over time as additional conventions are identified and past conventions are rendered obsolete by changes in the language itself.

Many projects have their own coding style guidelines. In the event of any conflicts, such project-specific guides take precedence for that project.

A Foolish Consistency is the Hobgoblin of Little Minds

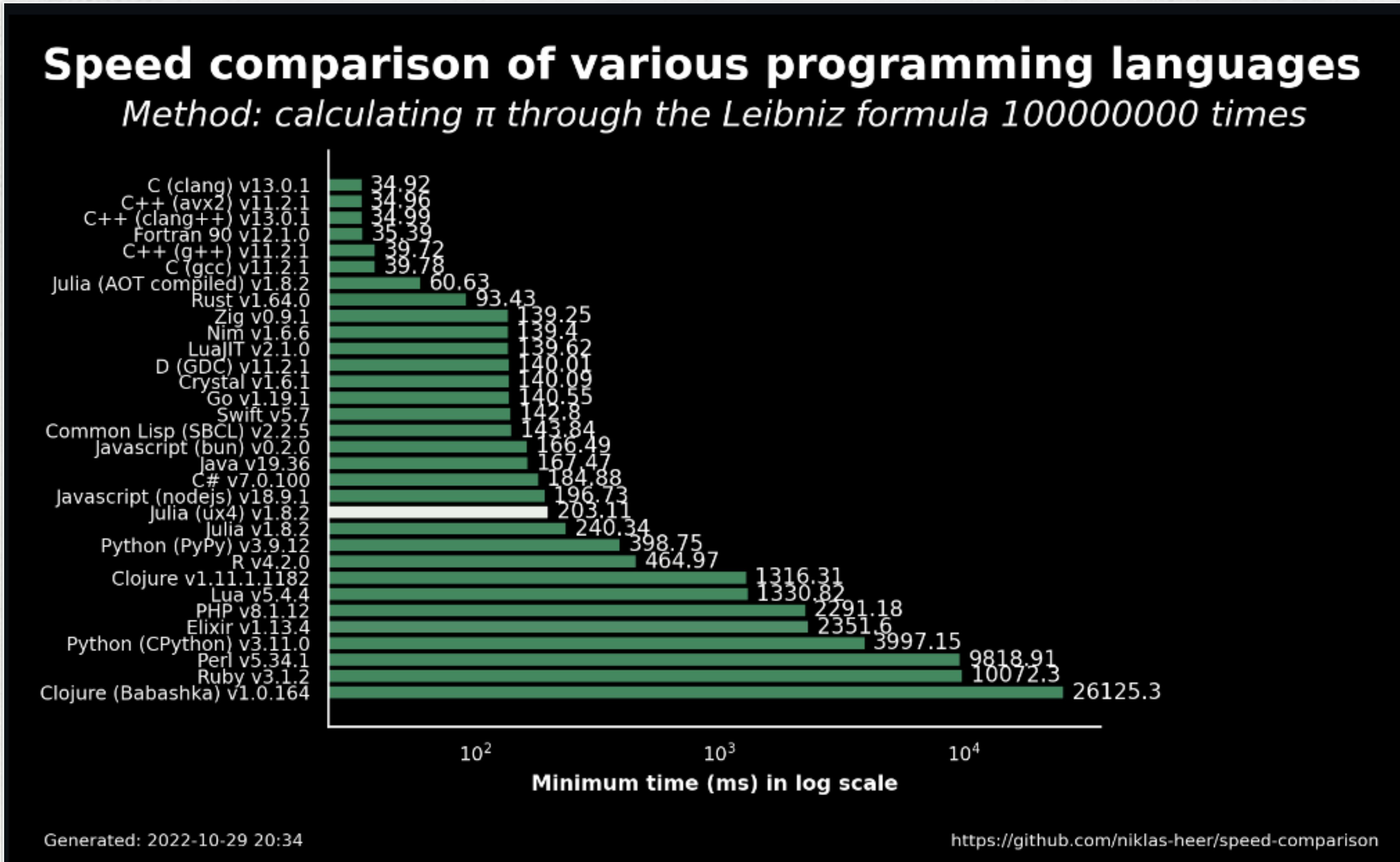
One of Guido's key insights is that code is read much more often than it is written. The guidelines provided here are intended to improve the readability of code and make it consistent across the wide spectrum of Python code. As [PEP 20](#) says, "Readability counts".

A style guide is about consistency. Consistency with this style guide is important. Consistency within a project is more important. Consistency within one module or function is the most important.

However, know when to be inconsistent – sometimes style guide recommendations just aren't applicable. When in doubt, use your best judgment. Look at other examples and decide what looks best. And don't hesitate to ask!

Python 성능 이해하기

- Speed comparison of programming languages
 - <https://github.com/niklas-heer/speed-comparison/blob/master/README.md>



Python 성능 이해하기

- How Slow is Python Compared to C

- <https://peter-jp-xie.medium.com/how-slow-is-python-compared-to-c-3795071ce82a>

**45,000 times
slower!**

Python 성능 개선하기

- PythonSpeed PerformanceTips
- <https://wiki.python.org/moin/PythonSpeed/PerformanceTips>

The screenshot shows a Python wiki page titled "PerformanceTips". The left sidebar includes links for FRONTPAGE, RECENTCHANGES, FINDPAGE, HELPCONTENTS, and PERFORMANCE TIPS (which is highlighted). The main content area has a "Contents" sidebar with a numbered list of performance tips from 1 to 16, each with a blue link. The main text area contains introductory text about the page's purpose, a note about Python changing over time, a warning about testing tips, and a mention of modern performance tools like Cython and PyPy.

This page is devoted to various tips and tricks that help improve the performance of your Python programs. Wherever the information comes from someone else, I've tried to identify the source.

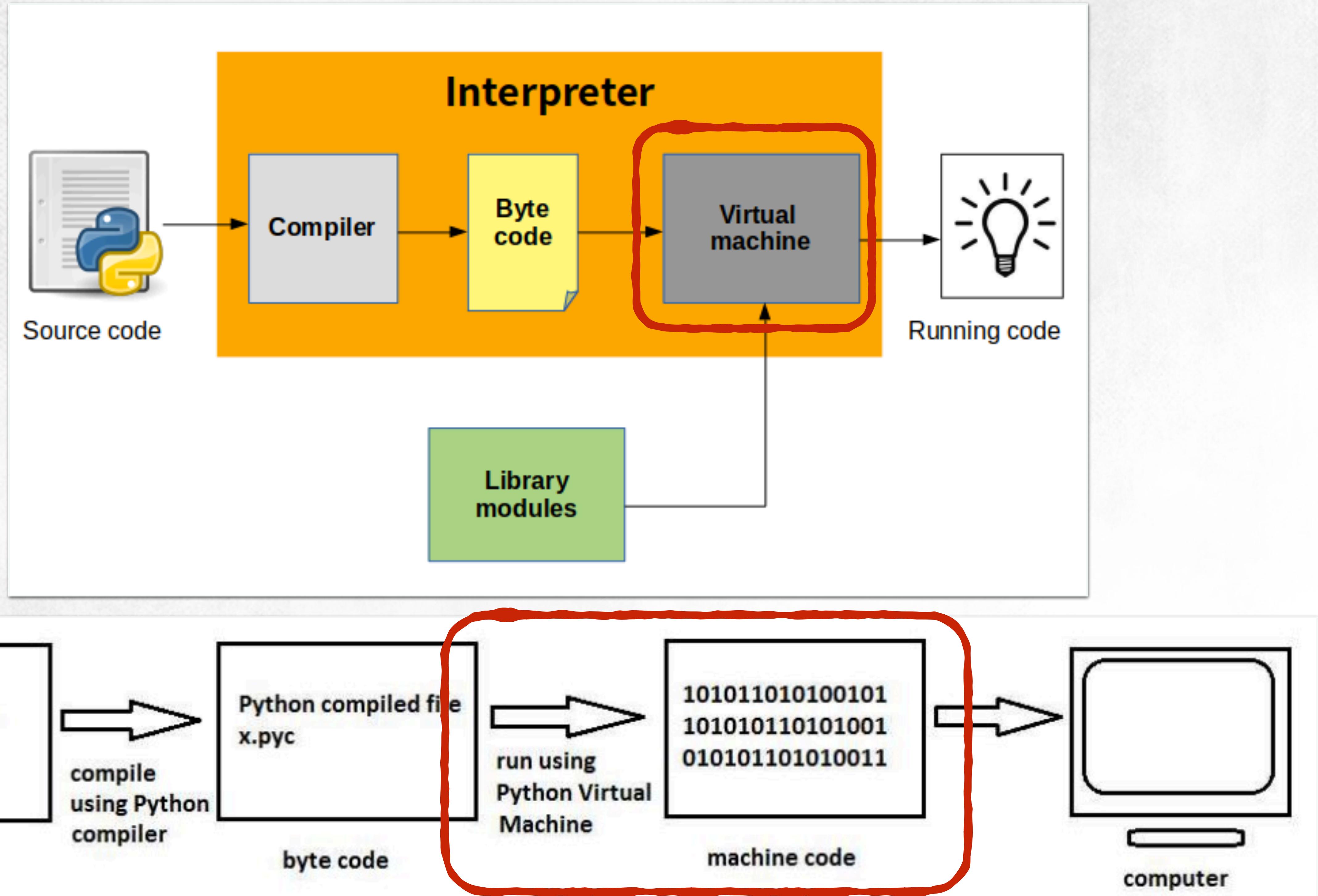
Python has changed in some significant ways since I first wrote my "fast python" page in about 1996, which means that some of the orderings will have changed. I migrated it to the Python wiki in hopes others will help maintain it.

You should always test these tips with your application and the specific version of the Python implementation you intend to use and not just blindly accept that one method is faster than another. See the [profiling](#) section for more details.

Also new since this was originally written are packages like [Cython](#), [Pyrex](#), [Psyco](#), [Weave](#), [Shed Skin](#) and [PyInline](#), which can dramatically improve your application's performance by making it easier to push performance-critical code into C or machine language.

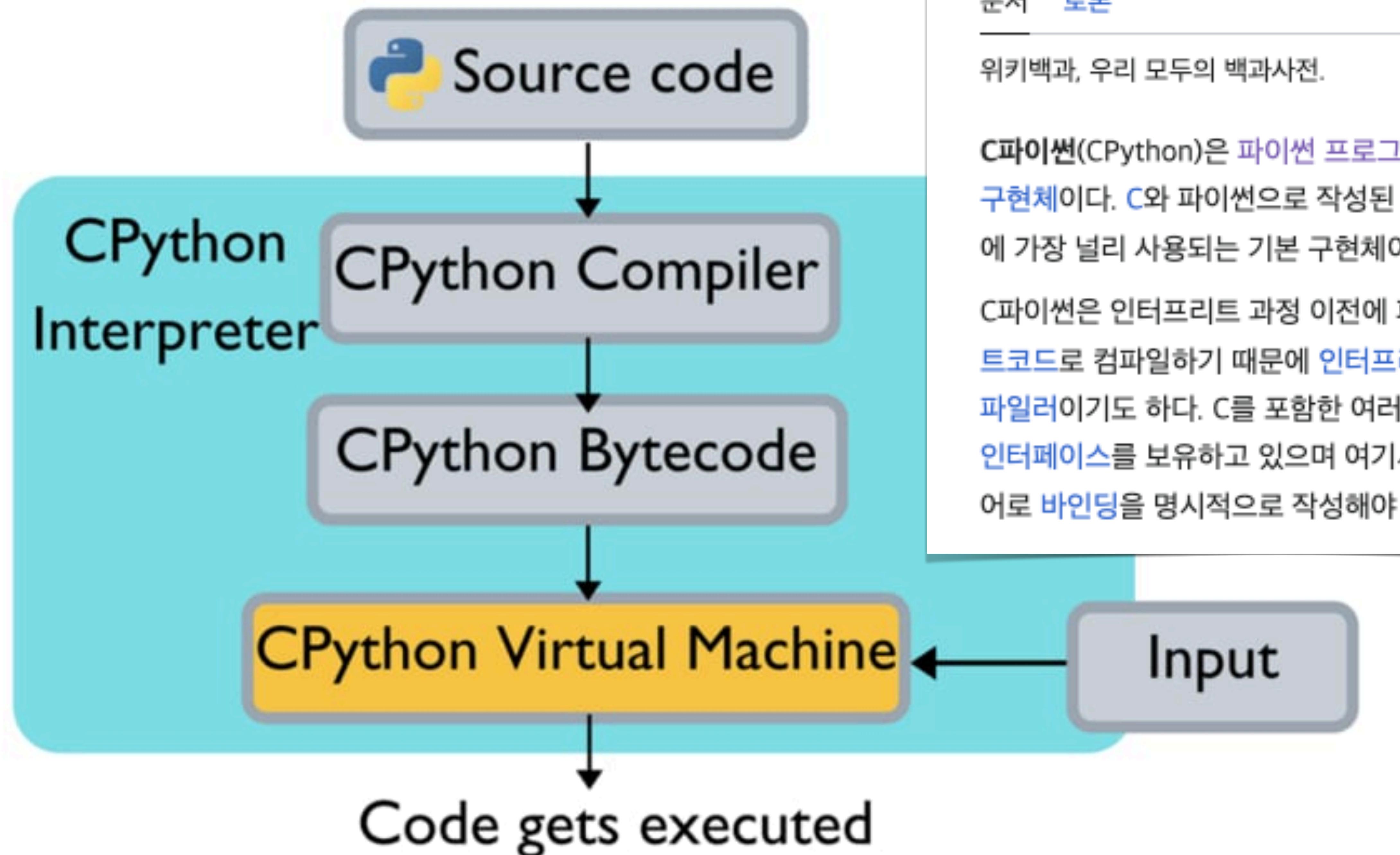
Python 성능 개선하기

Python Virtual Machine



Python 성능 개선하기

● CPython



C파이썬

문서 토론

위키백과, 우리 모두의 백과사전.

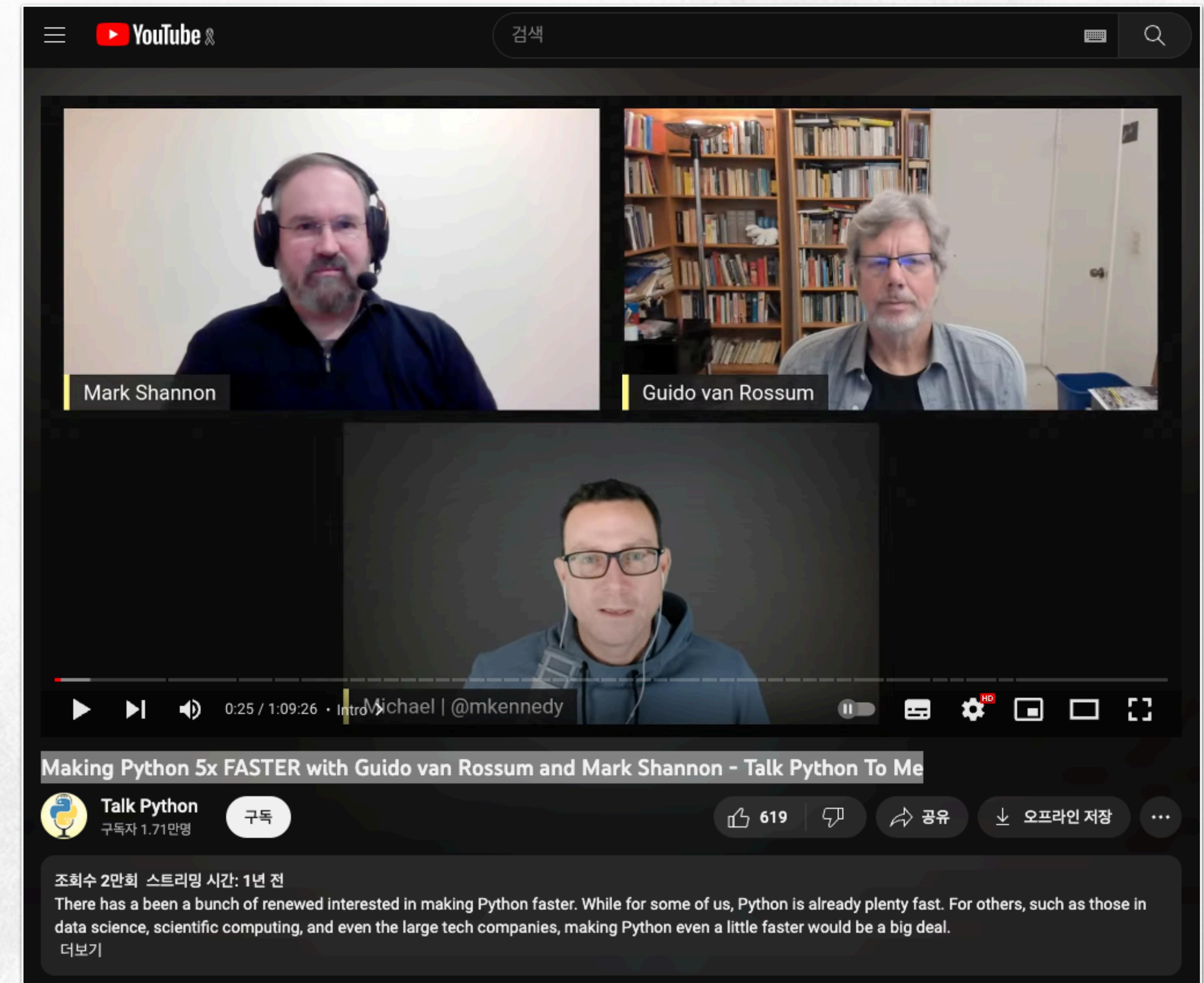
C파이썬(CPython)은 파이썬 프로그래밍 언어의 참조 구현체이다. C와 파이썬으로 작성된 C파이썬은 이 언어에 가장 널리 사용되는 기본 구현체이다.

C파이썬은 인터프리트 과정 이전에 파이썬 코드를 바이트코드로 컴파일하기 때문에 인터프리터이기도 하고 컴파일러이기도 하다. C를 포함한 여러 언어의 외부 함수 인터페이스를 보유하고 있으며 여기서 파이썬 외의 언어로 바인딩을 명시적으로 작성해야 한다.

Python 성능 개선하기

- Making Python 5x FASTER with **Guido van Rossum** and Mark Shannon - Talk Python To Me (2021)
- https://www.youtube.com/watch?v=_r6bFhl6wR8

Guido van Rossum
=
Father of Python

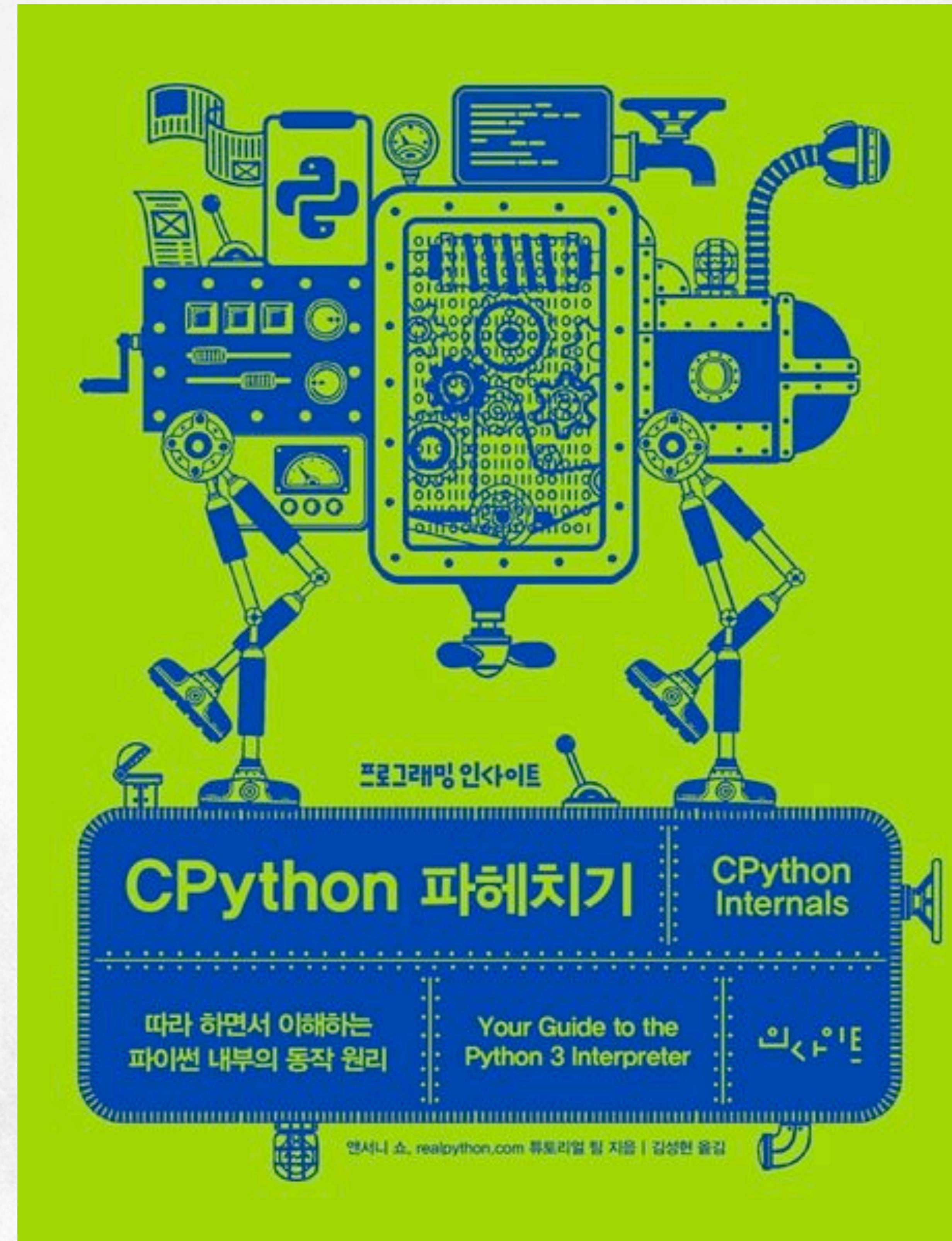


Python 성능 개선하기

- Faster CPython, 어떻게 더 빨라졌을까? 라인개발실록 (2022)
- <https://www.youtube.com/watch?v=KSNlslmwPyI&list=WL>



Python 성능 개선하기



새로운 Python 기술 학습하기

Python for Unity

- https://docs.unity3d.com/Packages/com.unity.scripting.python@2.0/manual/index.html

The screenshot shows the Unity Documentation website for Python for Unity 2.0.1-preview.2. The top navigation bar includes links for Manual, Scripting API, Changelog, and License, along with a search bar and the URL docs.unity3d.com.

The main content area displays the Python for Unity manual page. It features a sidebar with links to Python for Unity, Installation, and Using Python in Unity. The main content section is titled "Python for Unity" and discusses its benefits, such as automating scene assembly and integrating with other tools. It also mentions requirements and provides details about the Python Script Editor and various APIs.

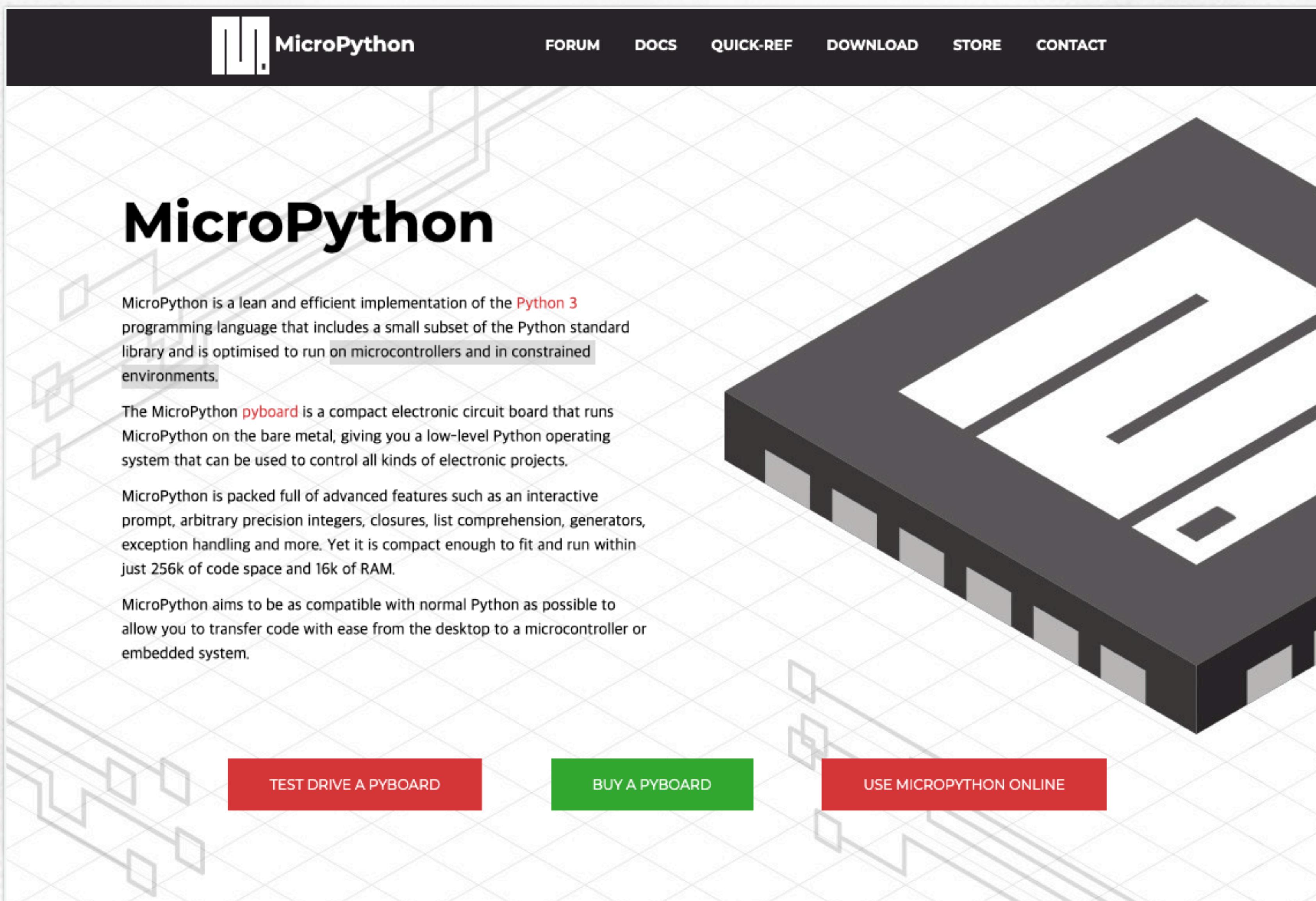
At the bottom, there's a rating section with five yellow stars and a link to report a problem.

Did you find this page useful? Please give it a rating:

Report a problem on this page

새로운 Python 기술 학습하기

- MicroPython : Python on microcontrollers and in constrained environments.
- <https://micropython.org/>



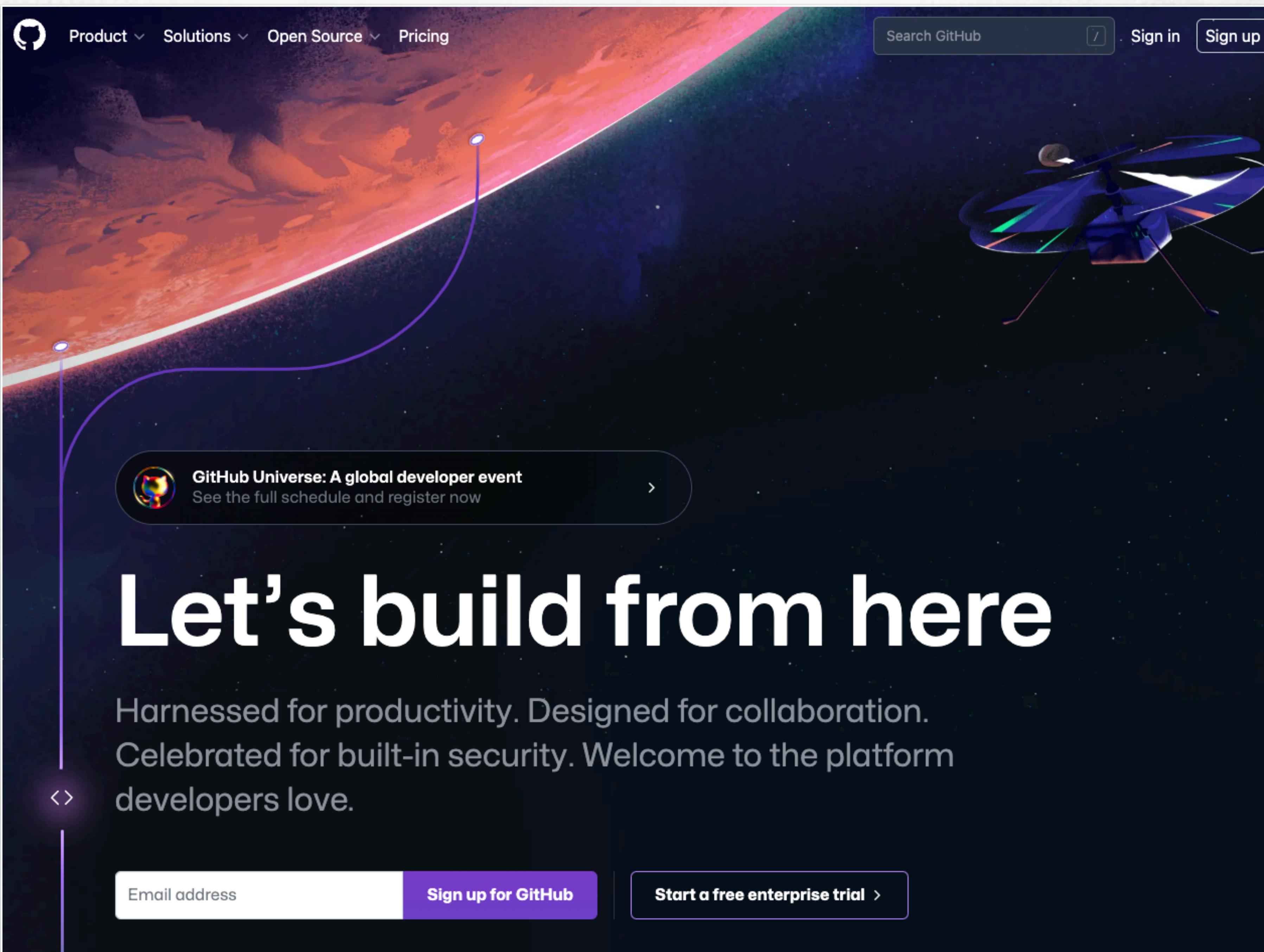
새로운 Python 기술 학습하기

- PyScript : Run Python in Your HTML
 - <https://pyscript.net/>



포트폴리오 사이트 구축하기

- GitHub : Software development maintenance site
- <https://github.com/>



Self-Motivated Programmer

대학의 목표는 인간의 자유와 성숙이지,
특정 업무를 수행할 개인을 만드는 것이 아니며,
후자는 오히려 노예제의 정의에 가깝다.

“대학의 배신”, 마이클로스



Thank you