

What's new in Python



Veit Schiele
Cusy GmbH, Berlin
Python Users Berlin, 13 May 2021

github.com/cusyio/jupyter-reveal/pub/pub_2021-05.pdf

Python is on Mars

LINUX Unplugged did an interview with Tim Canham, the Mars Helicopter Operations Lead.

From the 43 minute he talks about Python: [How Linux Got to Mars](#).

See also:

- github.com/nasa/fprime

Improved error handling

On 9 April, Pablo Galindo, release manager for Python 3.10 and 3.11, tweeted a question to Python trainers:

Python educators and users: I have been working on improving SyntaxError messages in CPython lately. What error (only SyntaxErrors for now 😊) messages you or your students have struggle with? Which ones you think we should improve? 🤔 (Pls RT to help reaching more people 🙏).

twitter.com/pyblogsal/status/1380516575485263873

Improved error handling

On 9 April, Pablo Galindo, release manager for Python 3.10 and 3.11, tweeted a question to Python trainers:

Python educators and users: I have been working on improving SyntaxError messages in CPython lately. What error (only SyntaxErrors for now 😊) messages you or your students have struggle with? Which ones you think we should improve? 🤔 (Pls RT to help reaching more people 🙏).

twitter.com/pyblogsal/status/1380516575485263873

He has received many answers also to improved error messages for assignment (=) and comparison (==) operators, indentation errors, and missing colons.

Improved error handling

On 9 April, Pablo Galindo, release manager for Python 3.10 and 3.11, tweeted a question to Python trainers:

Python educators and users: I have been working on improving SyntaxError messages in CPython lately. What error (only SyntaxErrors for now 😊) messages you or your students have struggle with? Which ones you think we should improve? 🤔 (Pls RT to help reaching more people 🙏).

twitter.com/pyblogsal/status/1380516575485263873

He has received many answers also to improved error messages for assignment (=) and comparison (==) operators, indentation errors, and missing colons.

See also:

- [Better error messages](#)

Python 3.8.10, 3.9.5, and 3.10.0b1 are now available

See also:

- [Python 3.8.10, 3.9.5, and 3.10.0b1 are now available](#)

Python 3.8.10, 3.9.5, and 3.10.0b1 are now available

Python 3.10.0b1

- Python 3.10 is now in Beta
- Beta release previews are intended to give the wider community the opportunity to test new features and bug fixes.
- Third-party Python projects should test and prepare their projects to support the new feature release.
- [PEP 619 – Python 3.10 Release Schedule](#)

Python 3.8.10, 3.9.5, and 3.10.0b1 are now available

Python 3.9.5

- [Python 3.9.5 Changelog](#)
- Python 3.9.5 is the newest major **stable** release.
- On macOS, you should use the `universal2` binary installer variant whenever possible. The legacy 10.9+ Intel-only variant will not be provided for Python 3.10 and the `universal2` variant will become the default download for future 3.9.x releases.

Python 3.8.10, 3.9.5, and 3.10.0b1 are now available

Python 3.8.10

- [PEP 569 – Python 3.8 Release Schedule](#)
- Python 3.8.10 is the final regular maintenance release.
- Starting now, the 3.8 branch will only accept security fixes.
- Releases will be made in source-only form until October 2024.

CPython repository switched from master to main

- Community's take on changing master branch to main
- github.com/github/renaming

Coverage.py

The next version of coverage.py will be 6.0, dropping support for Python 2 and Python 3.5.

See also:

- [Comparing changes](#)

PCA on the security incident disclosed by codecov

Codecov disclosed a security incident: [Bash Uploader Security Update](#).

The PCA (Python Cryptographic Authority) noted that this has [no impact on their use of codecov](#).

PyCon US 2021

<https://us.pycon.org/2021>

- Tutorials: 12–13 May 2021
- Conference: 14–15 May 2021
- Job Fair: 16 May 2021
- Sprints: 16–18 May 2021

Two times faster CPython in 3.11

Two times faster CPython in 3.11

Planning

- [The Shannon Plan](#)
- Based on experience with [HotPy](#)
- Small team funded by Microsoft with Eric Snow, Mark Shannon and Guido van Rossum
- Fully open collaboration with core devs
- Open [Issue tracker for Faster CPython project](#)
- [Tools for gathering bytecode statistics](#)
- [PEP 659 – Specializing Adaptive Interpreter](#)

Two times faster CPython in 3.11

Constraints

- ABI compatibility
- API compatibility
- Don't slow down extreme cases
- Keep code maintainable

Two times faster CPython in 3.11

How to

- Adaptive, specialising byte code interpreter
- Various other specialised optimisations
 - optimise frame stack
 - faster calls
 - tweak allocation
 - *Zero overhead* exception handling

Two times faster CPython in 3.11

Benefits

- CPU intensive pure Python Code
- Websites build in Python

Two times faster CPython in 3.11

Benefits

- CPU intensive pure Python Code
- Websites build in Python

Low benefits

- Code that's already in C
- I/O bound code
- Multi-threading code

Python Software Foundation

- [Call for nomination for the PSF board](#)

Nominations ends on 31 May 2021.

New in our bookshelf

- [Machine Learning mit Python](#) by Sebastian Raschka und Vahid Mirjalili
- [Refactoring](#) by Martin Fowler