You need a strategy for code quality.

Heather White · @eevelweezel

Quality Code…

- Does what it’s supposed to
- Can be easily tested with automated tests
- Follows a consistent style
- Is understandable
- Doesn't contain security vulnerabilities
- Is documented well
- Is easy to maintain

…and every developer has a different idea of what that looks like.

- A code quality strategy should cover your expectations for testing, linting, formatting, security, and maintainability
  - Bonus points if it also covers your docs
- For best results, you want to automate enforcement

Linters

- Flag code that’s incompatible with pep8 or your custom style guide
- Highly configurable
- …which leads to disagreements
  - Single vs double quotes, line length, spaces vs tabs…
- Examples: pylint, pyflakes, flake8, ruff

Formatters

- Reformat code to match the style guide
- Less configurable
  - black does not care if you prefer single quotes
- Grumbling vs. disagreements
- Examples: isort, black, ruff, prettier

Cyclomatic Complexity and Maintainability

- Assign a score representing code complexity
  - McCabe Complexity measures distinct code paths
  - Halstead Metrics are based on the total number of operators and operands (similar to the ideal gas law)
- Prevent your colleague from committing that class with multiple inheritance and 500+ lines of nested if’s
- Examples: flake8, ruff, radon

Security Vulnerability Scanners

- Check for unsafe practices or dependencies with known vulnerabilities
  - Beyond tools on PyPI, there are many commercial products available
- **Understand and configure your tool.**
  - Some tools will flag the GPL as a vulnerability
  - How does your tool handle disputed CVEs?
  - If builds don’t fail, it will be treated as noise
- Examples: bandit, safety, snyk

These tools need to run in order to provide value.

- In-editor linting
- Before committing to the codebase
- In CICD
- Make this mechanism as simple as possible
  - Hint: use pre-commit
  - Demo: let’s set up pre-commit for sparklepony

Further reading

PyCQA documentation: https://flake8.pycqa.org/en/latest/index.html

Pre-commit: https://pre-commit.com/

Featured pre-commit hooks: https://pre-commit.com/hooks.html

A disputed CVE filed against pip: https://www.cve.org/CVERecord?id=CVE-2018-20225

Ned Batchelder on the original mccabe script:

http://nedbatchelder.com/blog/200803/python_code_complexity_microtool.html

Wikipedia on Cyclomatic Complexity: https://en.wikipedia.org/wiki/Cyclomatic_complexity

Heather White (Weezel)

eevel.weezel@gmail.com
https://github.com/eevelweezel
https://salsa.debian.org/eevelweezel