
Digitise, Optimise, Visualise: Programming Style

Peter H. Gruber

July 1-5, 2019

So what's the goal?

Create working code with the least effort.

(An optimization problem)

Why good programming style?

The Goal
▷ Elements of style
Coding
Zen of Python
Style guides

Avoid errors.

- Save time (searching for errors)
- Save embarrassment (if you don't find them)

Memories.

- We usually overestimate how much we remember
- We usually underestimate the number of future changes

Collaboration.

- Common standard with co-authors, co-workers
- What happens if you change jobs?

Reproducibility.

- Document how every step from the data to your results
- Publish code to allow for reproducibility

Remember: Code is read by humans

4 Steps: Planning – coding – documenting – debugging

Phase 2: Coding

The Goal
Elements of style
▷ Coding
Zen of Python
Style guides

Have a reader on your mind. Would they understand your writing?

- ☐ Write clearly – don't be too clever.
- ☐ Choose the most direct representation of your problem.
- ☐ Use a standard code layout.
 - Preambles
 - Work with white spaces (equations, empty lines)
- ☐ Use library functions.
- ☐ Write and test a big program in small pieces.
- ☐ Use versioning.
- ☐ Adopt a naming convention.

PEP 20: The Zen of Python

01. Beautiful is better than ugly.
02. Explicit is better than implicit.
03. Simple is better than complex.
Complex is better than complicated.
05. Flat is better than nested.
06. Sparse is better than dense.
07. Readability counts.
08. Special cases aren't special enough to break the rules.
Although practicality beats purity.
10. Errors should never pass silently.
Unless explicitly silenced.
12. In the face of ambiguity, refuse the temptation to guess.
13. There should be one-- and preferably only one --obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
15. Now is better than never.
Although never is often better than **right** now.
17. If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
19. Namespaces are one honking great idea -- let's do more of those!

Style guides

The Goal
Elements of style
Coding
Zen of Python
▷ Style guides

- ☐ PEP 8 (=Python Enhancement Proposal 8)
<https://www.python.org/dev/peps/pep-0008/>
- ☐ Google Python Style Guide
<http://google.github.io/styleguide/pyguide.html>
- ☐ Hitchhiker's guide to Python style guide
<https://docs.python-guide.org/writing/style/>