Anna Roszman
Junior Backend Developer @ Wirtualna Polska
(WP Pilot, WP Program TV, Open FM)
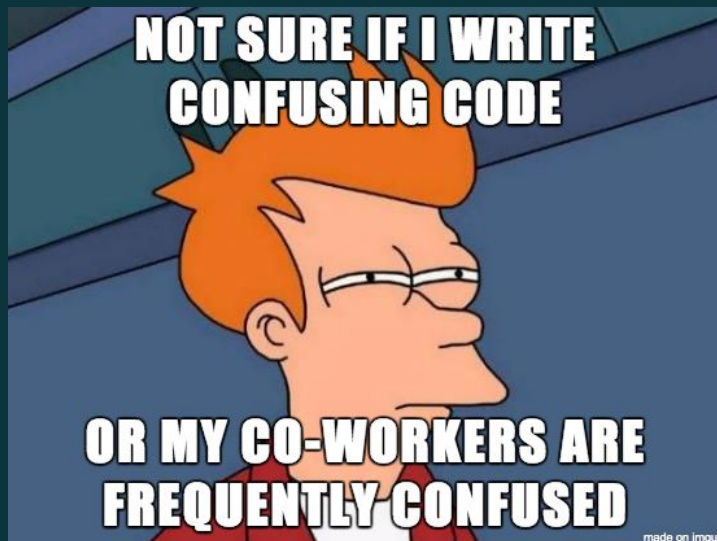
Dobre nawyki, czyli jak nie utrudniać sobie życia
Good practices, or how to make life easier

linkedin.com/in/anna-roszman
github.com/rosz

"Code is more often read than written."
Guido Van Rossum

PEP

Python Enhancement Proposals

github.com/python/peps

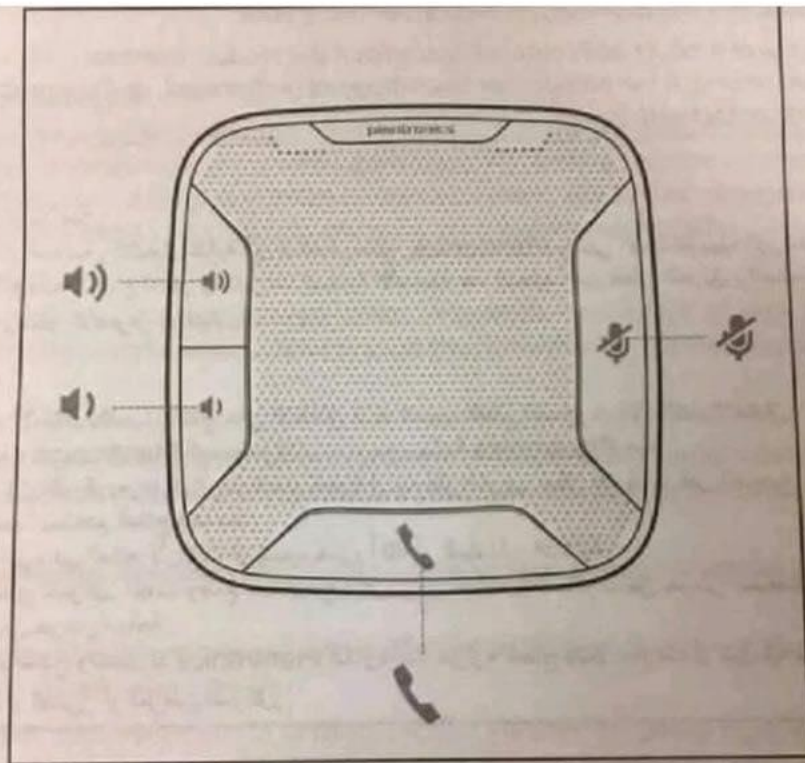# PEP 8

Style Guide for Python Code

PEP8:

>code layout (indentations, line breaking, blank lines, white spaces)

>imports (standard library, related third party imports, local imports)

>naming conventions (variable_names, function_names, ClassNames, CONSTANTS...; names to avoid)

>comments

>inheritance design

...and so on...

>project-specific guides take precedence for that project

How I comment my code.

comments vs docstrings

Comments:
>"how?" (not "what?")
>for programmers who develop the code
>purposes: planning, tagging, review; explanation of more complicated structures
>*comments that contradict the code are worse than no comments* (must be up-to-date)

Docstrings:
>"what?" (not "how?")
>summary
>for developers who use the code

```python
class JSONDecodeError(ValueError):
    """Subclass of ValueError with the following additional properties:

    msg: The unformatted error message
    doc: The JSON document being parsed
    pos: The start index of doc where parsing failed
    lineno: The line corresponding to pos
    colno: The column corresponding to pos

    """
    # Note that this exception is used from _json
    def __init__(self, msg, doc, pos):
        lineno = doc.count('\n', 0, pos) + 1
        colno = pos - doc.rfind('\n', 0, pos)
        errmsg = '%s: line %d column %d (char %d)' % (msg, lineno, colno, pos)
        ValueError.__init__(self, errmsg)
        self.msg = msg
        self.doc = doc
        self.pos = pos
        self.lineno = lineno
        self.colno = colno

    def __reduce__(self):
        return self.__class__, (self.msg, self.doc, self.pos)
```

```
In [1]: def do_nothing():
   ...:      """Do nothing.
   ...:
   ...:      Document it.
   ...:      """
   ...:      pass

In [2]: print(do_nothing.__doc__)
Do nothing.

    Document it.

In [3]: help(do_nothing)
Help on function do_nothing in module __main__:

do_nothing()
    Do nothing.

    Document it.
```
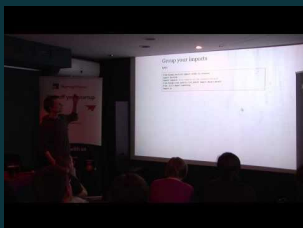
comments
vs.
self-documenting code

linter (lint)

>general name for tools that analyze source code

>flags errors, problematic constructs, etc.

>may be integrated with IDE, version control system (git) or development systems

>exceptions may (and should) be applied

>i.e. PyLint, Black, Flake8, mypy

More...

Agata Tyśnicka, PyLight #5
Czym jest refactoring i dlaczego jest ważny?

Ignacy Sokołowski, PyWaw #35
Readability counts

Wojtek Ebertowski, PyWaw #74
Dokumentacja w samodokumentującym się świecie

Raymond Hettinger, PyCon 2015
Beyond PEP 8 - best practices for beautiful intelligible code