

Zapachy kodu

Zapachy kodu

Cechy kodu mówiące o złym sposobie implementacji i będące sygnałem do refaktoryzacji.

Zapachy kodu

Najpopularniejsza obecnie klasyfikacja, wprowadzona przez Mika Mantyla w artykule: A Taxonomy for "Bad Code Smells" (2006 r.) wyróżnia pięć kategorii zapachów kodu.

Klasyfikacja Mantyla



Bloaters



Dispensables



Change Preventers



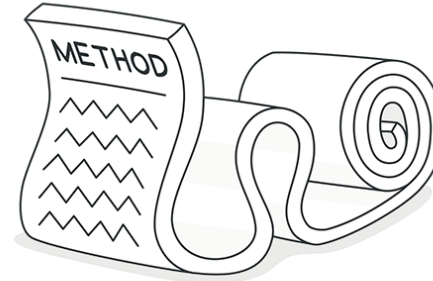
Object-Orientation Abuser



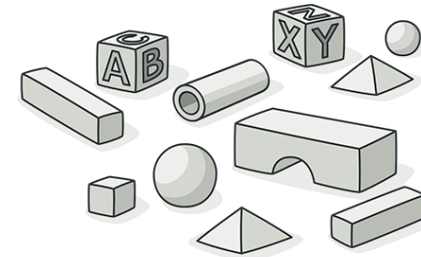
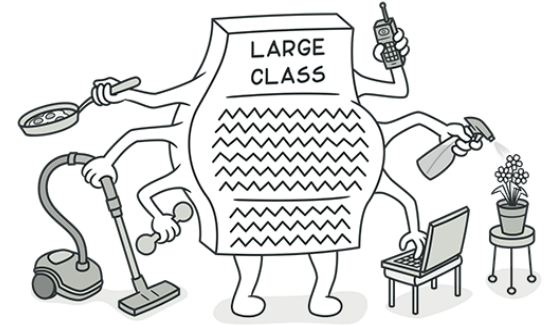
Couplers

Bloaters	Object-Orientation Abusers	Change Preventers	Dispensables	Couplers
<ul style="list-style-type: none">Long MethodLarge ClassPrimitive ObsessionLong Parameter ListData Clumps	<ul style="list-style-type: none">Alternative Classes with Different InterfaceRefused BequestSwitch StatementsTemporary Field	<ul style="list-style-type: none">Divergent ChangeParallel Inheritance HierarchiesShotgun Surgery	<ul style="list-style-type: none">CommentsDuplicate CodeData ClassDead CodeLazy ClassSpeculative Generality	<ul style="list-style-type: none">Feature EnvyInappropriate IntimacyIncomplete Library ClassMessage ChainsMiddle Man

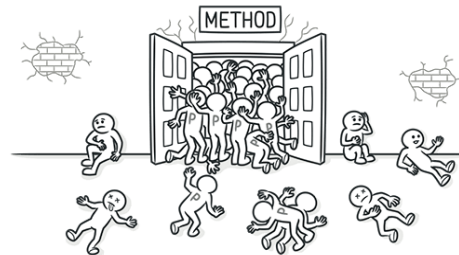
Bloaters



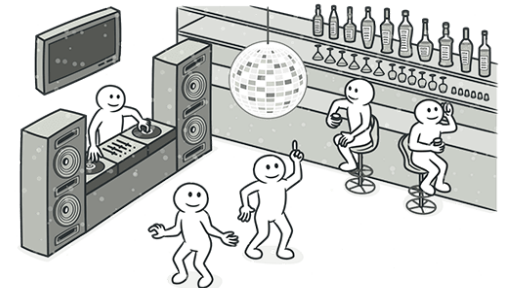
Long Method



Primitive Obsession

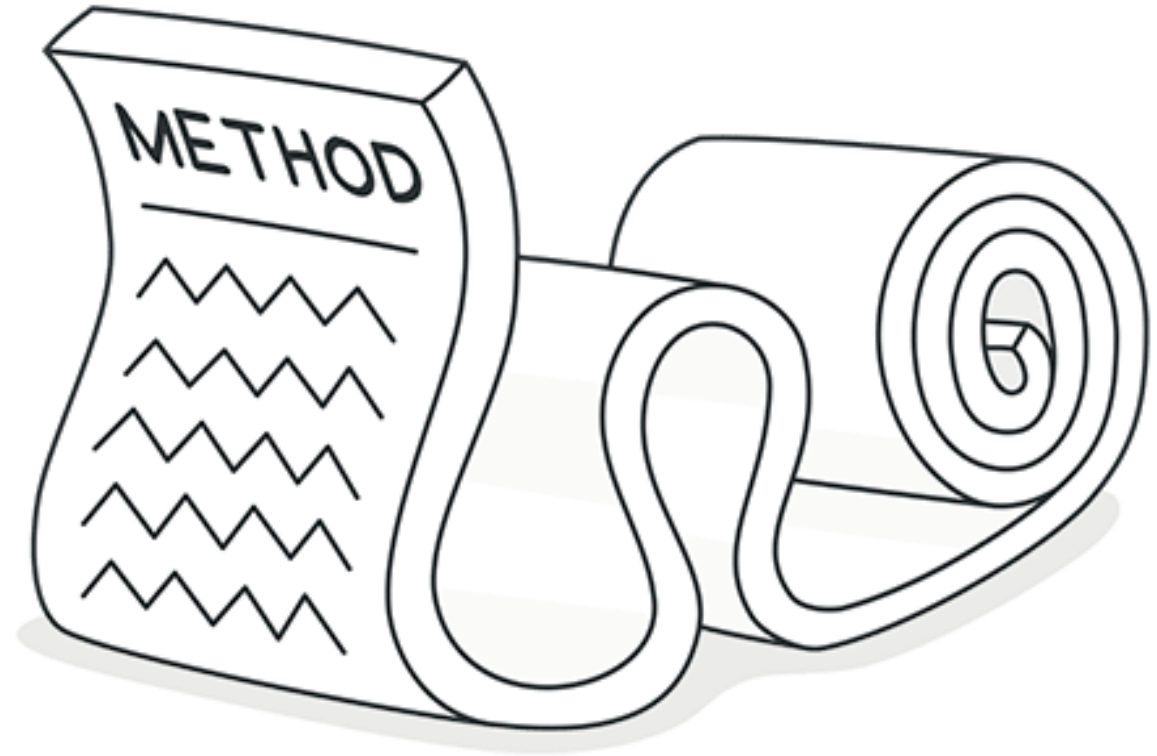


Long Parameter List

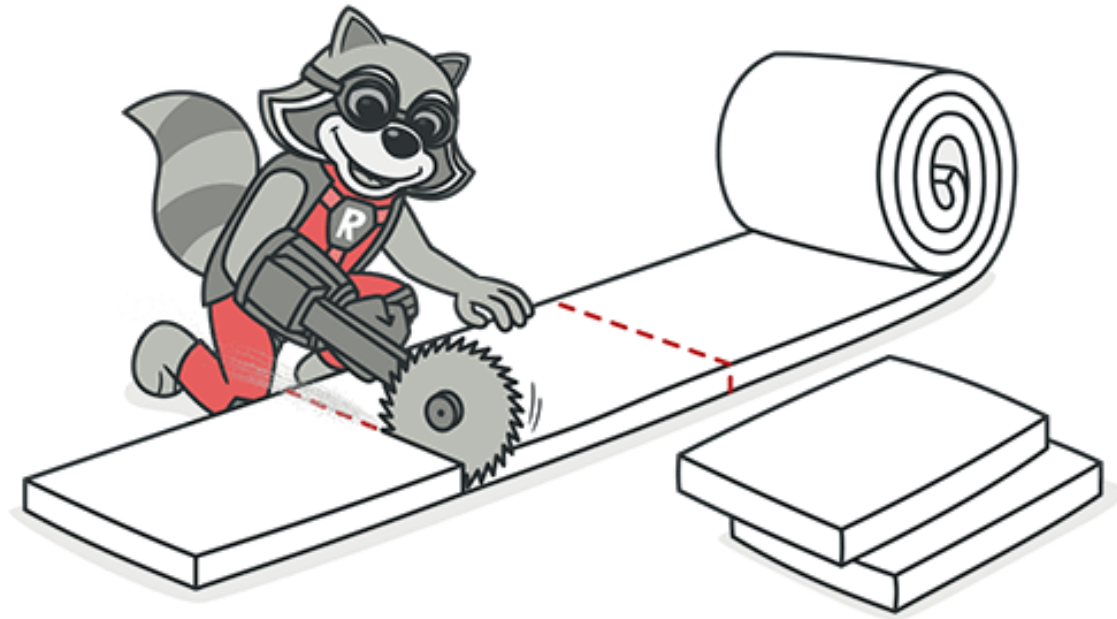


Data Clumps

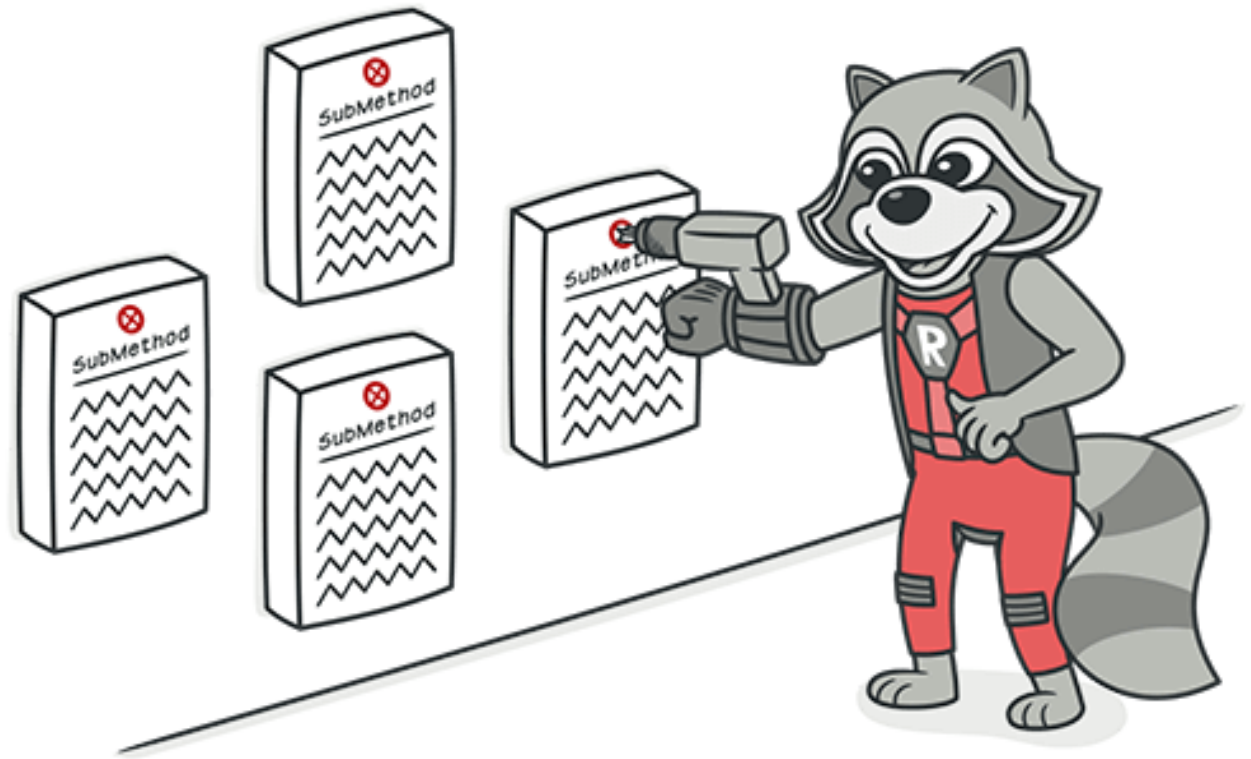
Long method



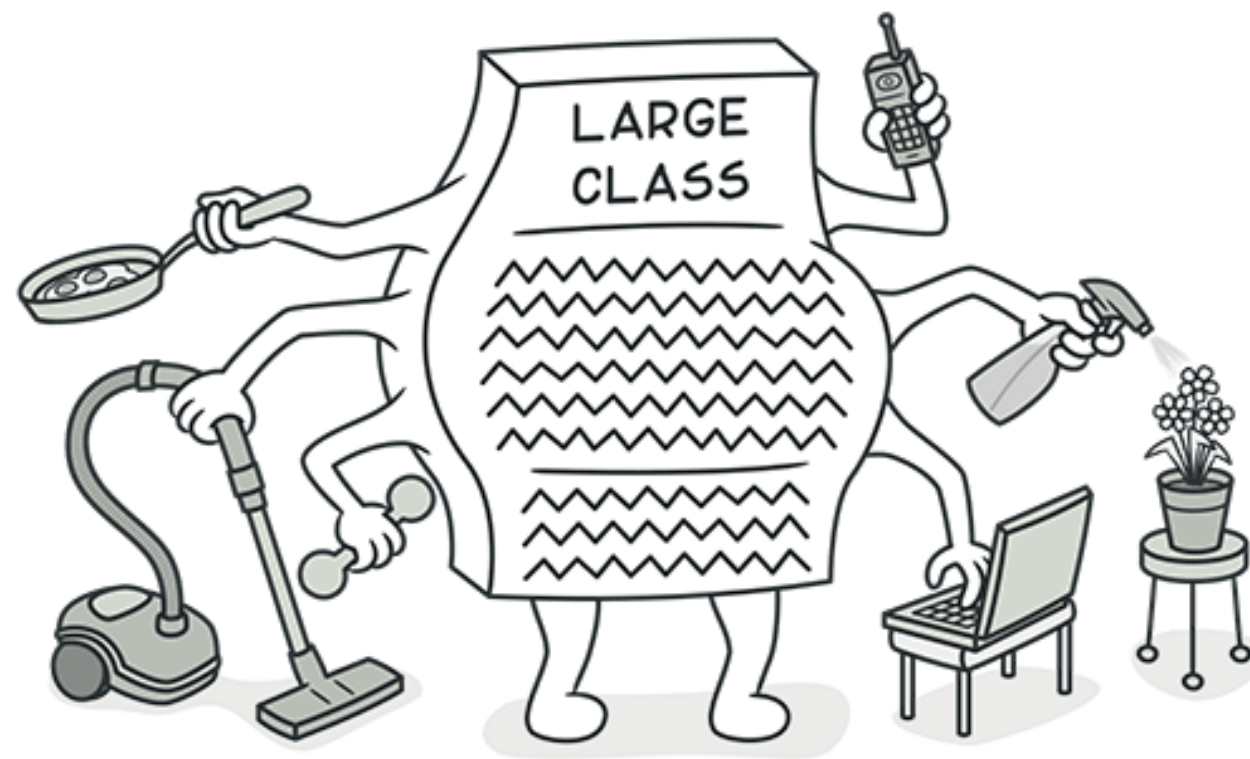
Long method



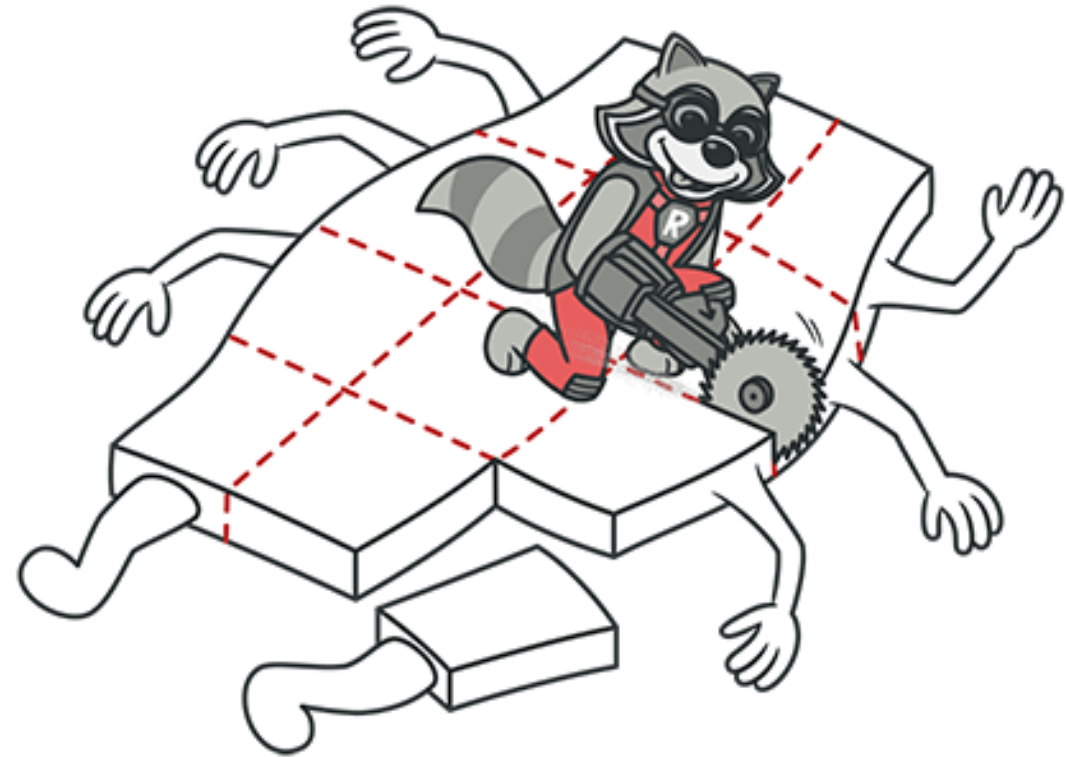
Long method



Large class



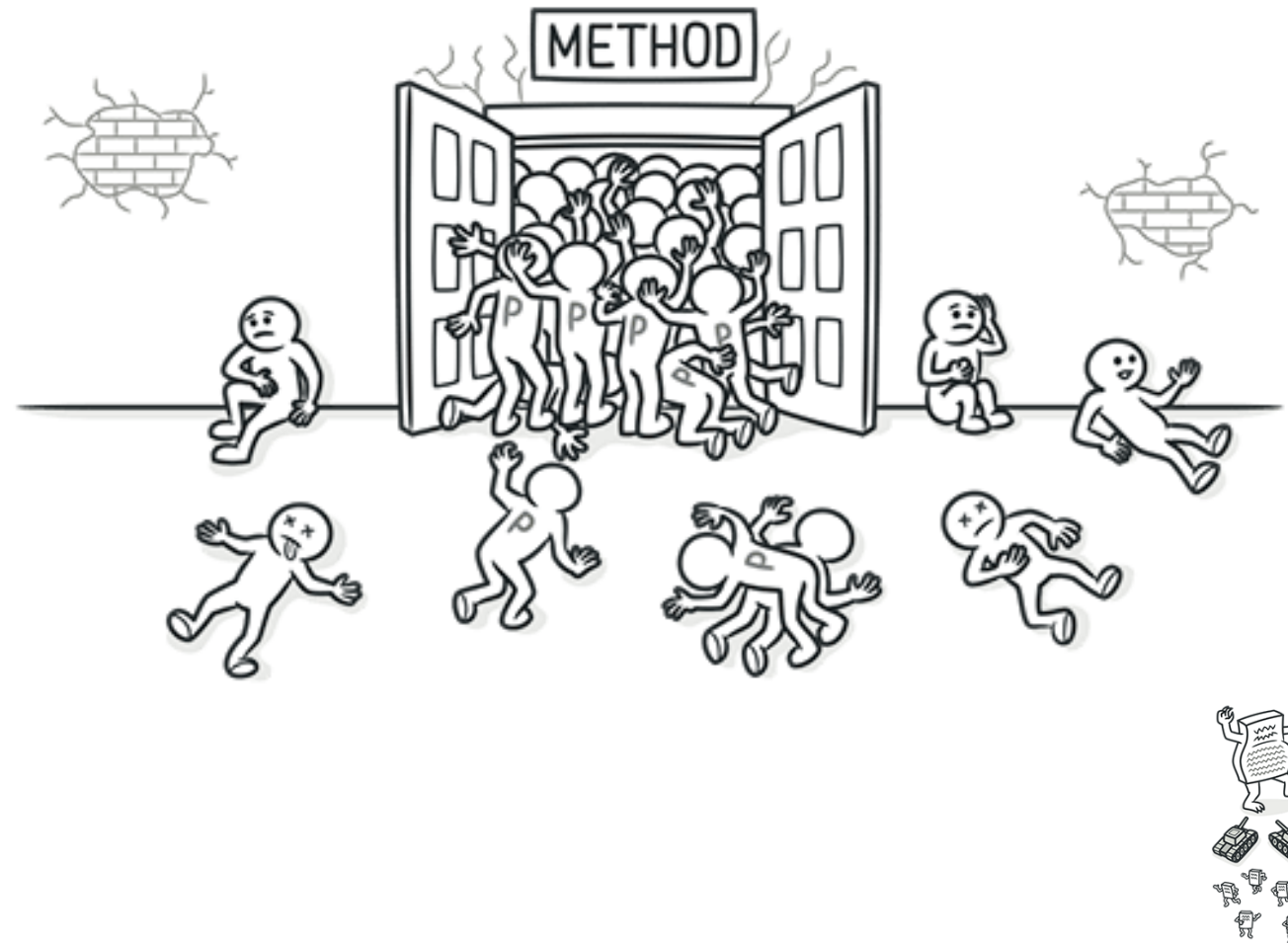
Large class



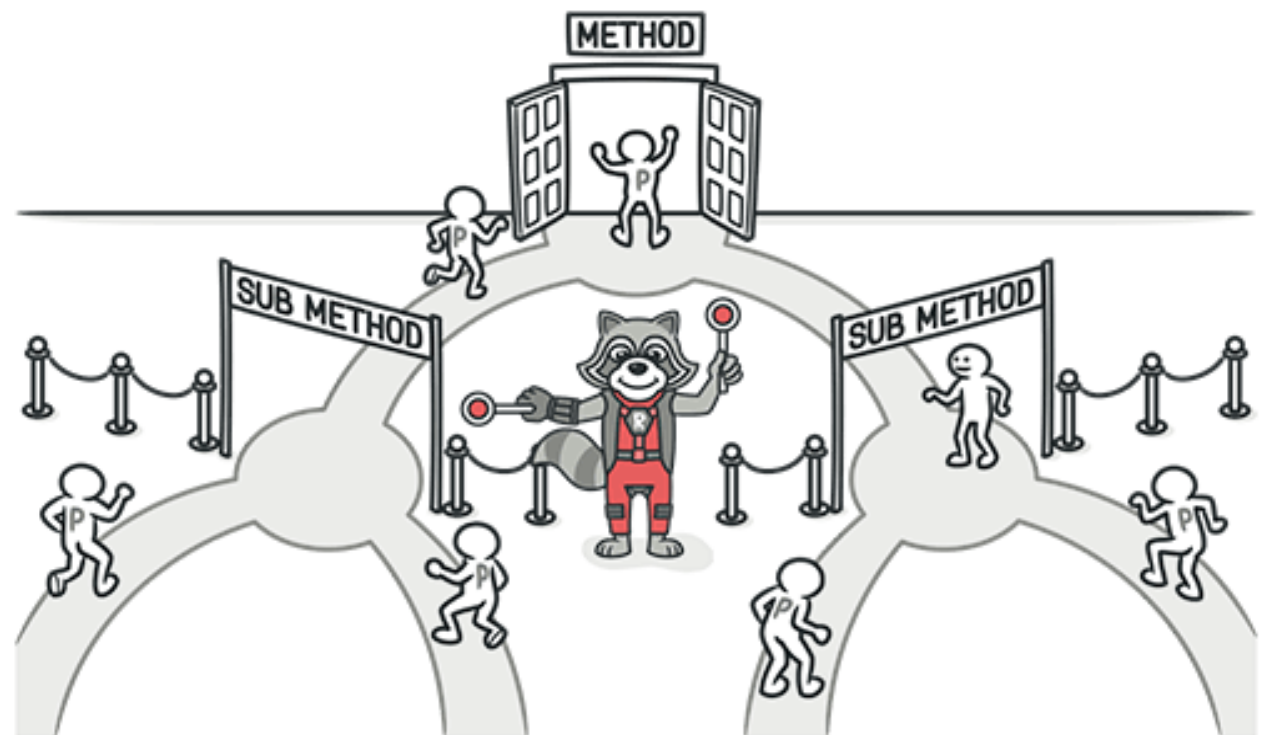
Large class



Long parameter list




Long parameter list



Dobre praktyki



PEP – Python Enhancement Proposals



PEP – Python
Enhancement Proposals

Co to PEP ?

PEP – Python Enhancement Proposals

Co to PEP ?

Zbiór oficjalnych, numerowanych dokumentów informacyjnych społeczność Pythona o wprowadzanych zmianach, propozycjach zmian, przyjętych konwencjach i standardach.

PEP 1

PEP – Python Enhancement Proposals

◇ <https://www.python.org/dev/peps>

PEP – Python Enhancement Proposals

- ◇ <https://www.python.org/dev/peps>
- ◇ PEP 8 – Style Guide for Python Code

PEP – Python Enhancement Proposals

- ◇ <https://www.python.org/dev/peps> - pełna lista
- ◇ PEP 8 – Style Guide for Python Code
- ◇ PEP 257 – Docstring Conventions

PEP – Python Enhancement Proposals

- ◇ <https://www.python.org/dev/peps>
- ◇ PEP 8 – Style Guide for Python Code
- ◇ PEP 257 – Docstring Conventions
- ◇ PEP 318 – Decorators for Functions and Methods

PEP 8



PEP 8

◆ Style Guide for Python Code:

PEP 8

- ◆ Style Guide for Python Code:
 - ◆ wcięcia za pomocą 4 spacji

PEP 8

- ◇ Style Guide for Python Code:
 - ◇ wcięcia za pomocą 4 spacji
 - ◇ maksymalna długość linii - 79

PEP 8

- ◇ Style Guide for Python Code:
 - ◇ wcięcia za pomocą 4 spacji
 - ◇ maksymalna długość linii - 79
 - ◇ odległości w kodzie (białe linie i spacje)

PEP 8

- ◇ Style Guide for Python Code:
 - ◇ wcięcia za pomocą 4 spacji
 - ◇ maksymalna długość linii - 79
 - ◇ odległości w kodzie (białe linie i spacje)
 - ◇ komentarze

PEP 8

- ◇ Style Guide for Python Code:
 - ◇ wcięcia za pomocą 4 spacji
 - ◇ maksymalna długość linii - 79
 - ◇ odległości w kodzie (białe linie i spacje)
 - ◇ komentarze
 - ◇ nazewnictwo zmiennych i funkcji – snake_case

PEP 8

- ◇ Style Guide for Python Code:
 - ◇ wcięcia za pomocą 4 spacji
 - ◇ maksymalna długość linii - 79
 - ◇ odległości w kodzie (białe linie i spacje)
 - ◇ komentarze
 - ◇ nazewnictwo zmiennych i funkcji – snake_case
 - ◇ nazewnictwo klas – CamelCase

Google Python Style Guide

