

- Secrets
- ABAP
- Apex
- C
- C++
- CloudFormation
- COBOL
- C#
- CSS
- Flex
- Go
- HTML
- Java
- JavaScript
- Kotlin
- Objective C
- PHP
- PL/I
- PL/SQL
- Python**
- RPG
- Ruby
- Scala
- Swift
- Terraform
- Text
- TypeScript
- T-SQL
- VB.NET
- VB6
- XML



Python static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your PYTHON code

All rules 216

Vulnerability 29

Bug 55

Security Hotspot 31

Code Smell 101

Tags ▾

Search by name...



Functions should not have too many lines of code

Code Smell

Track uses of "NOSONAR" comments

Code Smell

Track comments matching a regular expression

Code Smell

Statements should be on separate lines

Code Smell

Functions should not contain too many return statements

Code Smell

Files should not have too many lines of code

Code Smell

Lines should not be too long

Code Smell

Methods and properties that don't access instance data should be static

Code Smell

New-style classes should be used

Code Smell

Parentheses should not be used after certain keywords

Code Smell

Track "TODO" and "FIXME" comments that do not contain a reference to a person

Code Smell

Module names should comply with a naming convention

Loops without "break" should not have "else" clauses

Analyze your code

Code Smell Major suspicious

The else clause of a loop is skipped when a break is executed in this loop. In other words, a loop with an else but no break statement will always execute the else part (unless of course an exception is raised or return is used). If this is what the developer intended, it would be much simpler to have the else statement removed and its body unindented. Thus having a loop with an else and no break is most likely an error.

Noncompliant Code Example

```
from typing import List

def search_first_number_without_break(elements: List[str])
    for elt in elements:
        if elt.isnumeric():
            return elt
    else: # Noncompliant. This will be executed every
        raise ValueError("List does not contain any num
```

Compliant Solution

```
from typing import List

def search_first_number_with_break(elements: List[str])
    for elt in elements:
        if elt.isnumeric():
            break
    else:
        raise ValueError("List does not contain any num
    return elt
```






or

```
from typing import List

def search_first_number_without_else(elements: List[str])
    for elt in elements:
        if elt.isnumeric():
            return elt
    raise ValueError("List does not contain any number"
```

See

- [Python documentation - break and continue Statements, and else Clauses on Loops](#)

 Code Smell
Comments should not be located at the end of lines of code  Code Smell
Lines should not end with trailing whitespaces  Code Smell
Files should contain an empty newline at the end  Code Smell
Long suffix "L" should be upper case  Code Smell

Available In:

sonarlint  | **sonarcloud**  | **sonarqube** 

© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved.
[Privacy Policy](#)