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...

☢ Code Smell

"SystemExit" should be re-raised

☢ Code Smell

Bare "raise" statements should only be used in "except" blocks

☢ Code Smell

Comparison to None should not be constant

☢ Code Smell

"self" should be the first argument to instance methods

☢ Code Smell

Function parameters' default values should not be modified or assigned

☢ Code Smell

Some special methods should return "NotImplemented" instead of raising "NotImplementedError"

☢ Code Smell

Custom Exception classes should inherit from "Exception" or one of its subclasses

☢ Code Smell

Bare "raise" statements should not be used in "finally" blocks

☢ Code Smell

Arguments given to functions should be of an expected type

☢ Code Smell

`str.replace` should be preferred to `re.sub`

☢ Code Smell

Unread "private" attributes should be removed

The "exec" statement should not be used

Analyze your code

☢ Code Smell   🚫 Blocker ?   🏷 python3 obsolete

Use of the exec statement could be dangerous, and should be avoided. Moreover, the exec statement was removed in Python 3.0. Instead, the built-in exec() function can be used.

Noncompliant Code Example

```
exec 'print 1' # Noncompliant
```

Compliant Solution

```
exec('print 1')
```

Available In:

sonarlint 💬 | sonarcloud 🔗 | sonarqube 📶

Code Smell

Cognitive Complexity of functions should not be too high

Code Smell

The first argument to class methods should follow the naming convention

Code Smell

Method overrides should not change contracts

Code Smell

Wildcard imports should not be used

Code Smell