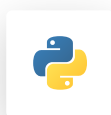


- Secrets
- ABAP
- Apex
- C
- C++
- CloudFormation
- COBOL
- C#
- CSS
- Flex
- Go
- HTML
- Java
- JavaScript
- Kotlin
- Objective C
- PHP
- PL/I
- PL/SQL
- Python**
- RPG
- Ruby
- Scala
- Swift
- Terraform
- Text
- TypeScript
- T-SQL
- VB.NET
- VB6
- XML



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...



Security Hotspot

Reading the Standard Input is security-sensitive

Security Hotspot

Using command line arguments is security-sensitive

Security Hotspot

Encrypting data is security-sensitive

Security Hotspot

Using regular expressions is security-sensitive

Security Hotspot

Dynamically executing code is security-sensitive

Security Hotspot

Cyclomatic Complexity of functions should not be too high

Code Smell

Control flow statements "if", "for", "while", "try" and "with" should not be nested too deeply

Code Smell

Cyclomatic Complexity of classes should not be too high

Code Smell

"\" should only be used as an escape character outside of raw strings

Bug

Using shell interpreter when executing OS commands is security-sensitive

Security Hotspot

Functions should use "return" consistently

Related "if/else if" statements should not have the same condition

Analyze your code

Bug Major ? unused pitfall

A chain of `if/else if` statements is evaluated from top to bottom. At most, only one branch will be executed: the first one with a condition that evaluates to `true`.

Therefore, duplicating a condition automatically leads to dead code. Usually, this is due to a copy/paste error. At best, it's simply dead code and at worst, it's a bug that is likely to induce further bugs as the code is maintained, and obviously it could lead to unexpected behavior.

Noncompliant Code Example

```
if param == 1:
    openWindow()
elif param == 2:
    closeWindow()
elif param == 1:                # Noncompliant
    moveWindowToTheBackground()
```

Compliant Solution






```
if param == 1:
    openWindow()
elif param == 2:
    closeWindow()
elif param == 3:
    moveWindowToTheBackground()
```

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](#)

 Code Smell
Python parser failure  Code Smell
Files should not be too complex  Code Smell
Docstrings should be defined  Code Smell
Functions should not have too many lines of code  Code Smell
Track uses of "NOSONAR" comments