

Objective C

PHP

PL/I

Oii

- PL/SQL
- **Pvthon**
- **RPG**
- A Ruby
- Scala
- Swift
- Terraform
- Text
- **TypeScript**
- T-SQL
- **VB.NET**
- VB<sub>6</sub>
- **XML**



## Python static code analysis

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



Tags

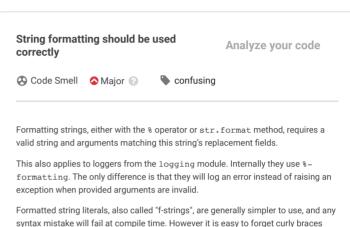


person

Code Smell

naming convention A Code Smell

Module names should comply with a



Search by name...

and it won't raise any error This rule raises an issue when:

- A string formatted with % will not return the expected string because some arguments are not used.
- A string formatted with str.format will not return the expected string because some arguments are not used.
- An "f-string" doesn't contain any replacement field, which probably means that some curly braces are missing.
- Loggers will log an error because their message is not formatted properly.

Rule {rule:python:S2275} covers cases where formatting a string will raise an

## **Noncompliant Code Example**

```
"Error %(message)s" % {"message": "something failed", "extra
"Error: User {} has not been able to access []".format("Alic
user = "Alice"
resource = "MyFile"
message = f"Error: User [user] has not been able to access [
logging.error("Error: User %s has not been able to access %s
```

## **Compliant Solution**

```
"Error %(message)s" % {"message": "something failed"}
"Error: User {} has not been able to access {}".format("Alic
user = "Alice'
resource = "MyFile"
message = f"Error: User {user} has not been able to access {
import logging
logging.error("Error: User %s has not been able to access %s
```

Comments should not be located at the end of lines of code

A 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

## See

- Python documentation Format String Syntax
- Python documentation printf-style String Formatting
- Python documentation Loggers
- Python documentation Using particular formatting styles throughout your application
- Python documentation Formatted string literals

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