



C# C#

€ CSS **X** Flex

-GO Go

5 HTML

Java Java Java Script

Kotlin

Kubernetes

Objective C

PHP

PL/I PL/I

PL/SQL PL/SQL

Python

RPG RPG

Ruby

Scala

Swift

Terraform

Text

Ts TypeScript

T-SQL

VB VB.NET

VB6 VB6

XML XML



Go static code analysis

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

Hard-coded credentials are securitysensitive Security Hotspot **Cognitive Complexity of functions** should not be too high Code Smell String literals should not be duplicated Code Smell Functions should not be empty Code Smell All branches in a conditional structure should not have exactly the same implementation 📆 Bug "=+" should not be used instead of "+=" 📆 Bug Related "if/else if" statements should not have the same condition 📆 Bug Identical expressions should not be used on both sides of a binary operator 📆 Bug All code should be reachable Rug Bug Variables should not be self-assigned

📆 Bug

implementations

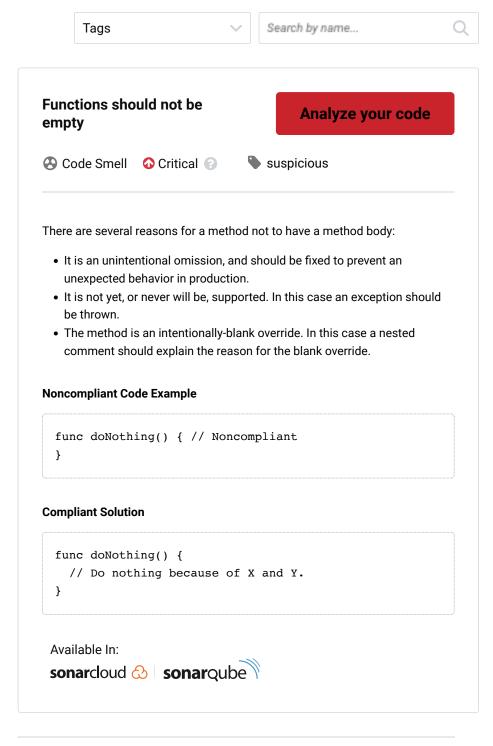
Code Smell

Code Smell

Functions should not have identical

Two branches in a conditional structure should not have exactly the

same implementation



© 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

"switch" statements should not have too many "case" clauses
Code Smell
Track uses of "FIXME" tags
Redundant pairs of parentheses should be removed
☼ Code Smell
Nested blocks of code should not be left empty
Code Smell
Functions should not have too many parameters
Code Smell
Using hardcoded IP addresses is security-sensitive
Security Hotspot
Multi-line comments should not be empty
Code Smell
Boolean checks should not be inverted
☼ Code Smell
Local variable and function parameter names should comply with a naming convention
☼ Code Smell
Boolean literals should not be redundant
Code Smell