

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your PL/SQL code

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your PL/SQL code

All rules 188

🔓 Vulnerability ④

 Bug (45)

 Security Hotspot 2

Code Smell 137

Tags

Search by name...

 Code Smell

Files should not have too many lines of code

 Code Smell

Lines should not be too long

 Code Smell

Explicitly opened cursors should be closed

 Bug

Identifiers should be written in lower case

 Code Smell

"PLS_INTEGER" types should be used

 Code Smell

Reserved words should be written in upper case

 Code Smell

Parameter "IN" mode should be specified explicitly

 Code Smell

Lines in a multiline comment should start with `"*`

 Code Smell

CASE should be used for sequences of simple tests

 Code Smell

SQL tables should be joined with the "JOIN" keyword

 Code Smell

Constraint names should comply with a naming convention

 Code Smell

Reserved words should be written in lower case

Files should not have too many lines of code

Analyze your code

 Code Smell
  Major
 
 brain-overload

A source file that grows too much tends to aggregate too many responsibilities and inevitably becomes harder to understand and therefore to maintain. Above a specific threshold, it is strongly advised to refactor it into smaller pieces of code which focus on well defined tasks. Those smaller files will not only be easier to understand but also probably easier to test.

Available In:

sonarlint  | sonarcloud  | sonarqube  Developer Edition

© 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](#)

Privacy Policy