





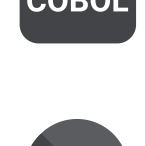



























-  Secrets
-  ABAP
-  Apex
-  C
-  C++
-  CloudFormation
-  COBOL
-  C#
-  CSS
-  Flex
-  Go
-  HTML
-  Java
-  JavaScript
-  Kotlin
-  Kubernetes
-  Objective C
-  PHP
-  PL/I
-  **PL/SQL**
-  Python
-  RPG
-  Ruby
-  Scala
-  Swift
-  Terraform
-  Text
-  TypeScript
-  T-SQL
-  VB.NET
-  VB6
-  XML

PL/SQL

PL/SQL static code analysis

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


Magic numbers should not be used

 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

Magic numbers should not be used

 Code Smell

 Major ?

 brain-overload


A magic number is a number that comes out of nowhere, and is directly used in a statement. Magic numbers are often used, for instance to limit the number of iterations of a loops, to test the value of a property, etc.


Using magic numbers may seem obvious and straightforward when you're writing a piece of code, but they are much less obvious and straightforward at debugging time.


That is why magic numbers must be demystified by first being assigned to clearly named variables before being used.

By default, -1, 0 and 1 are not considered magic numbers.

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](#)