


















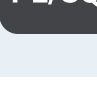














-  Secrets
-  ABAP
-  Apex
-  C
-  C++
-  CloudFormation
-  COBOL
-  C#
-  CSS
-  Flex
-  Go
-  HTML
-  Java
-  JavaScript
-  Kotlin
-  Kubernetes
-  Objective C
-  PHP
-  PL/I
-  **PL/SQL**
-  Python
-  RPG
-  Ruby
-  Scala
-  Swift
-  Terraform
-  Text
-  TypeScript
-  T-SQL
-  VB.NET
-  VB6
-  XML

## PL/SQL 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...

🔍

"IF" statements should not be nested too deeply

 Code Smell

"CASE" expressions should end with "ELSE" clauses

 Code Smell

String literals should not be duplicated

 Code Smell

Constant names should comply with a naming convention

 Code Smell

Output parameters should be assigned

 Bug

"ROWNUM" should not be used at the same query level as "ORDER BY"

 Bug

All branches in a conditional structure should not have exactly the same implementation

 Bug

Strings should only be moved to variables or columns which are large enough to hold them

 Bug

"WHERE" clause conditions should not be contradictory

 Bug

Unary prefix operators should not be repeated

 Bug

DML events clauses should not include multiple "OF" clauses

 Bug

"PACKAGE BODY" initialization sections should not contain "RETURN" statements

 Bug

"NUMBER" type should not be used in a function

"IF" statements should not be nested too deeply

 Code Smell

 Critical



 brain-overload

Nested IF statements are a key ingredient for making what's known as "Spaghetti code".

Such code is hard to read, refactor and therefore maintain.

Available In:

 |  |  Developer Edition

Analyze your code