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...

Anchored types should not be constrained
Bug

"DELETE" and "UPDATE" statements should contain "WHERE" clauses
Bug

"FORALL" statements should use the "SAVE EXCEPTIONS" clause
Bug

Pipelined functions should have at least one "PIPE ROW" statement and not return an expression (PLS-00633)
Bug

GOTO should not be used to jump backwards
Code Smell

Quoted identifiers should not be used
Code Smell

Loop start and end labels should match
Code Smell

Block start and end labels should match
Code Smell

Cipher algorithms should be robust
Vulnerability

"COMMIT" should not be used inside a loop
Bug

Individual "WHERE" clause conditions should not be unconditionally true or false
Bug

Nullable subqueries should not be used in "NOT IN" conditions
Bug

"COMMIT" and "ROLLBACK" should not be called from non-autonomous transaction triggers
Bug

Constant declarations should contain initialization assignments

Analyze your code

Bug Blocker

Constants must be immediately initialized at declaration. They cannot be reassigned any value after the declaration, as they are constant. This rule prevents PLS-00322 exceptions from being raised at runtime.

The following code snippet illustrates this rule:

Noncompliant Code Example

```
DECLARE
  foo CONSTANT PLS_INTEGER NULL; -- Noncompliant PLS-00322
  bar CONSTANT PLS_INTEGER NOT NULL; -- Noncompliant PLS-00322
  aa CONSTANT PLS_INTEGER; -- Noncompliant
BEGIN
  NULL;
END;
/
```

Compliant Solution

```
DECLARE
  foo CONSTANT PLS_INTEGER NULL :=42;
  bar CONSTANT PLS_INTEGER NOT NULL := 42;
  aa CONSTANT PLS_INTEGER := 42; -- Compliant
BEGIN
  NULL;
END;
/
```

Available In:
sonarlint sonarcloud sonarqube Developer Edition