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

Boolean checks should not be inverted
Code Smell

Unused local variables should be removed
Code Smell

Package names should comply with a naming convention
Code Smell

Variables should comply with a naming convention
Code Smell

Return of boolean expressions should not be wrapped into an "if-then-else" statement
Code Smell

Boolean literals should not be redundant
Code Smell

Comments should not be nested
Code Smell

"DBMS_UTILITY.FORMAT_ERROR_STACK" and "FORMAT_ERROR_BACKTRACE" should be used together
Code Smell

Procedures should not contain "RETURN" statements
Code Smell

Track uses of "TODO" tags
Code Smell

Neither DES (Data Encryption Standard) nor DESede (3DES) should be used
Vulnerability

Boolean checks should not be inverted

Analyze your code

Code Smell Minor pitfall

It is needlessly complex to invert the result of a boolean comparison. The opposite comparison should be made instead.

Noncompliant Code Example

```
IF NOT x <> y THEN   -- Noncompliant
  -- ...
END IF;
```

Compliant Solution

```
IF x = y THEN
  -- ...
END IF;
```

Available In:
sonarlint sonarcloud sonarqube Developer Edition