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...

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

Positional and named arguments should not be mixed in invocations
Bug

Functions should end with "RETURN" statements
Bug

Collections should not be iterated in "FOR" loops
Bug

Using weak hashing algorithms is security-sensitive
Security Hotspot

Dynamically executing code is security-sensitive
Security Hotspot

SQL "JOIN" conditions should involve all joined tables
Code Smell

Loop start and end labels should match

Analyze your code

Code Smell | Blocker | confusing

Labeled loops are useful, especially when the code is badly indented, to match the begin and end of each loop. This rule verifies that loop start and end labels match, when both are specified.

Noncompliant Code Example

```
BEGIN
  LOOP
    EXIT;
  END LOOP; -- Compliant, this loop has no label at all

  <<myLoopLabel1>>
  LOOP
    EXIT;
  END LOOP; -- Compliant, this loop only has a start label

  LOOP
    EXIT;
  END LOOP myLoopLabel2; -- Compliant, this loop only has an end label

  <<myLoopLabel4>>
  LOOP
    EXIT;
  END LOOP myLoopLabel5; -- Noncompliant, label mismatch

  <<myLoopLabel6>>
  <<myLoopLabel7>>
  LOOP
    EXIT;
  END LOOP myLoopLabel7; -- Noncompliant, several start labels mismatch
END;
/
```

Compliant Solution

```
BEGIN
  LOOP
    EXIT;
  END LOOP;

  <<myLoopLabel1>>
  LOOP
    EXIT;
  END LOOP;

  LOOP
    EXIT;
  END LOOP myLoopLabel2;

  <<myLoopLabel4>>
  LOOP
    EXIT;
  END LOOP myLoopLabel4;

  <<myLoopLabel7>>
  LOOP
    EXIT;
  END LOOP myLoopLabel7;
END;
/
```

Available In:

sonarlint | sonarcloud | sonarqube Developer Edition