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

Statements should be on separate lines
Code Smell

"WHEN" clauses should not have too many lines
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

Statements should be on separate lines

Analyze your code

Code Smell | Major | convention

For better readability, do not put more than one statement on a single line.

Noncompliant Code Example

```
SET SERVEROUTPUT ON

BEGIN
  DBMS_OUTPUT.PUT_LINE('Hello!'); DBMS_OUTPUT.PUT_LINE('This is unreadable!'); -- Noncompliant
END;
/
```

Compliant Solution

```
SET SERVEROUTPUT ON

BEGIN
  DBMS_OUTPUT.PUT_LINE('Hello!');
  DBMS_OUTPUT.PUT_LINE('This is much better!');
END;
/
```

Available In:

sonarlint | sonarcloud | sonarqube Developer Edition