








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...

Unused labels should be removed  Code Smell	
Compound triggers should define at least two triggers  Code Smell	
"EXIT" should not be used in loops  Code Smell	
Jump statements should not be redundant  Code Smell	
"EXCEPTION WHEN ... THEN" clauses should do more than "RAISE"  Code Smell	
Single line comments should start with "--"  Code Smell	
An "ORDER BY" direction should be specified explicitly  Code Smell	
Oracle's join operator (+) should not be used Code Smell	
"cursor%NOTFOUND" should be used instead of "NOT cursor%FOUND" Code Smell	
Object attributes should comply with a naming convention Code Smell	
Record fields should comply with a naming convention Code Smell	
Cursors should follow a naming convention Code Smell	

Unused labels should be removed

Analyze your code

 Code Smell
  Major
 
 unused

If a label is declared but not used in the program, it can be considered as dead code and should therefore be removed.

This will improve maintainability as developers will not wonder what this label is used for.

Noncompliant Code Example

```
<<foo>> -- Noncompliant
BEGIN
    DBMS_OUTPUT.PUT_LINE('Hello, world!');
END;
```

Compliant Solution

```
-- Compliant
BEGIN
    DBMS_OUTPUT.PUT_LINE('Hello, world!');
END;
```

See

- **CERT, MSC12-C.** - Detect and remove code that has no effect or is never executed

Available In:

sonarlint  | **sonarcloud**  | **sonarqube**  Developer Edition