





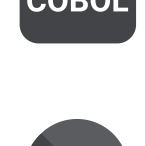



























-  Secrets
-  ABAP
-  Apex
-  C
-  C++
-  CloudFormation
-  COBOL
-  C#
-  CSS
-  Flex
-  Go
-  HTML
-  Java
-  JavaScript
-  Kotlin
-  Kubernetes
-  Objective C
-  PHP
-  PL/I
-  **PL/SQL**
-  Python
-  RPG
-  Ruby
-  Scala
-  Swift
-  Terraform
-  Text
-  TypeScript
-  T-SQL
-  VB.NET
-  VB6
-  XML

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 |
| Exceptions should follow a naming convention |
|  Code Smell |
| Exceptions should not be ignored |
|  Code Smell |
| Variables should not be initialized with "NULL" |
|  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 |
| ... |

Exceptions should follow a naming convention

Analyze your code

 Code Smell

 Minor 

 convention

Naming conventions allow teams to collaborate effectively. This rule checks that exception names match a given regular expression.

Noncompliant Code Example

With the default regular expression `[a-zA-Z] ([a-zA-Z0-9_] * [a-zA-Z0-9]) ? :`

```
DECLARE
    my-Exception_ EXCEPTION; -- Noncompliant
BEGIN
    NULL;
END;
/
```

Compliant Solution

```
DECLARE
    myException EXCEPTION;
BEGIN
    NULL;
END;
/
```

Available In:

sonarlint 

sonarcloud 

sonarqube  Developer Edition