

































-  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...



 Bug

Size should be specified for string variables

 Bug

Predefined exceptions should not be overridden

 Bug

"NOT NULL" variables should be initialized

 Bug

"NCHAR" and "NVARCHAR2" size should not be specified in bytes

 Bug

"END LOOP" should be followed by a semicolon

 Bug

Constraints should not be applied to types that cannot be constrained

 Bug

Improper constraint forms should not be used

 Bug

Scale should not be specified for float types

 Bug

Constant declarations should contain initialization assignments

 Bug

"RAISE_APPLICATION_ERROR" should only be used with error codes from -20,000 to -20,999

 Bug

"FETCH ... BULK COLLECT INTO" should not be used without a "LIMIT" clause

 Bug

Anchored types should not be constrained

 Bug

"DELETE" and "UPDATE" statements should contain "WHERE" clauses

Variables should be declared only once in a scope

Analyze your code

 Bug

 Blocker



At most one declaration of a variable in a given scope is allowed in PL/SQL. The PLS-00371 error will be raised at runtime when attempting to reference a variable declared more than once.

Noncompliant Code Example

```
SET SERVEROUTPUT ON

DECLARE
    foo VARCHAR2(42) := 'foo';
    foo VARCHAR2(42) := 'bar'; -- Non-Compliant
BEGIN
    DBMS_OUTPUT.PUT_LINE(foo); -- Raises PLS-00371: at most one declaration for 'FOO' is permitted
END;
/
```

Compliant Solution

```
SET SERVEROUTPUT ON

DECLARE
    foo VARCHAR2(42) := 'foo';
    bar VARCHAR2(42) := 'bar'; -- Compliant
BEGIN
    DBMS_OUTPUT.PUT_LINE(foo);
END;
/
```

Available In:

sonarlint

|

sonarcloud

|

sonarqube

Developer Edition