

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

Procedures should not contain "RETURN" statements

 Code Smell

Track uses of "TODO" tags

 Code Smell

Neither DES (Data Encryption Standard) nor DESede (3DES) should be used

Vulnerability

"SYNCHRONIZE" should not be used

 Bug

Global public variables should not be defined

 Code Smell

A primary key should be specified during table creation

 Code Smell

Track lack of copyright and license headers

Code Smell

SHA-1 and Message-Digest hash algorithms should not be used in secure contexts

Vulnerability

Sensitive "SYS" owned functions should not be used

Vulnerability

"FORMS_DDL('COMMIT')" and
"FORMS_DDL('ROLLBACK')" should not be used

 Bug

"DBMS_OUTPUT.PUT_LINE" should not be used

Code Smell

Procedures should not contain "RETURN" statements

Analyze your code

 Code Smell
  Minor
 
 brain-overload

Procedures, unlike functions, do not return values. The `RETURN` statement therefore, when used within a procedure, is used to prematurely end the procedure. However, having multiple exit points (i.e. more than the `END` of the procedure itself), increases the complexity of the procedure and makes it harder to understand and debug.

Noncompliant Code Example

```
SET SERVEROUTPUT ON

DECLARE

PROCEDURE prcoedureWithReturn AS
BEGIN
    RETURN; -- Noncompliant

    DBMS_OUTPUT.PUT_LINE('prcoedureWithReturn called'); -- This is actually unreachable
END;

BEGIN
    prcoedureWithReturn;
END;

/
```

Available In:

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