













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	
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
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

Return of boolean expressions should not be wrapped into an "if-then-else" statement

Analyze your code

 Code Smell

 Minor

 clumsy

Return of boolean literal statements wrapped into `if-then-else` ones should be simplified.


Noncompliant Code Example


```
IF expression THEN
    RETURN TRUE;
ELSE
    RETURN FALSE;
END IF;
```


Compliant Solution

```
RETURN expression;
```

Available In:

 sonarlint

 sonarcloud

 sonarqube

Developer Edition