





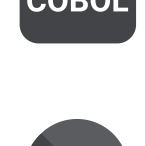



























-  Secrets
-  ABAP
-  Apex
-  C
-  C++
-  CloudFormation
-  COBOL
-  C#
-  CSS
-  Flex
-  Go
-  HTML
-  Java
-  JavaScript
-  Kotlin
-  Kubernetes
-  Objective C
-  PHP
-  PL/I
-  **PL/SQL**
-  Python
-  RPG
-  Ruby
-  Scala
-  Swift
-  Terraform
-  Text
-  TypeScript
-  T-SQL
-  VB.NET
-  VB6
-  XML

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... 🔍

Identifiers should be written in lower case
 Code Smell
"PLS_INTEGER" types should be used
 Code Smell
Reserved words should be written in upper case
 Code Smell
Parameter "IN" mode should be specified explicitly
 Code Smell
Lines in a multiline comment should start with "⌘"
 Code Smell
CASE should be used for sequences of simple tests
 Code Smell
SQL tables should be joined with the "JOIN" keyword
 Code Smell
Constraint names should comply with a naming convention
 Code Smell
Reserved words should be written in lower case
 Code Smell
Comments should not be located at the end of lines of code
 Code Smell
Lines should not end with trailing whitespaces
 Code Smell
The "RELIES_ON" clause should not be used
 Code Smell

## Constraint names should comply with a naming convention

Analyze your code

 Code Smell

 Minor ?

 convention

Sharing some naming conventions is a key factory in efficient team collaboration. This rule checks that all constraint names match a provided regular expression.

### Noncompliant Code Example

With `format_primary_key` set to `"pk_[a-z]++"` and `format_foreign_key` to `"fk_[a-z]++"`.

```
CREATE TABLE employee(  
    first_name VARCHAR2(42),  
    last_name VARCHAR2(42)  
    department_id INTEGER CONSTRAINT department_fk REFERENCES department,  
    CONSTRAINT fullname_pk PRIMARY KEY (first_name, last_name);  
);
```

### Compliant Solution

```
CREATE TABLE employee(  
    first_name VARCHAR2(42),  
    last_name VARCHAR2(42)  
    department_id INTEGER CONSTRAINT fk_department REFERENCES department,  
    CONSTRAINT pk_fullname PRIMARY KEY (first_name, last_name);  
);
```

Available In:

sonarlint 

sonarcloud 

sonarqube  Developer Edition