

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

Object attributes should comply with a naming convention

Code Smell

Record fields should comply with a naming convention

Code Smell

Cursors should follow a naming convention

Code Smell

Types should follow a naming convention

Code Smell

Cursor parameters should follow a naming convention

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

Object attributes should comply with a naming convention

Analyze your code

Code Smell

Minor

convention

Shared coding conventions allow teams to collaborate efficiently. This rule checks that object attribute names match the provided regular expression.

Noncompliant Code Example

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

```
CREATE TYPE my_type AS OBJECT(  
    foo_bar INTEGER                -- Non-Compliant  
);  
/  
  
DROP TYPE my_type;
```

Compliant Solution

```
CREATE TYPE my_type AS OBJECT(  
    foo_bar INTEGER                -- Compliant  
);  
/  
  
DROP TYPE my_type;
```

Available In:

sonarlint

sonarcloud

sonarqube

Developer Edition