sonar RULES

Products

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

Java static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your JAVA code

All rules 632 | Vulnerability 53 | Bug 154 | Security Hotspot 36 | Code Smell 389 | Quick Fix 42

Tags

Search by name...

A field should not duplicate the name of its containing class
Code Smell

JUnit4 @Ignored and JUnit5 @Disabled annotations should be used to disable tests and should provide a rationale
Code Smell

Anonymous inner classes containing only one method should become lambdas
Code Smell

"switch" statements should not have too many "case" clauses
Code Smell

"for" loop stop conditions should be invariant
Code Smell

Sections of code should not be commented out
Code Smell

Non-constructor methods should not have the same name as the enclosing class
Code Smell

Exception types should not be tested using "instanceof" in catch blocks
Code Smell

Classes from "sun.*" packages should not be used
Code Smell

Throwable and Error should not be caught
Code Smell

Unused method parameters should be removed
Code Smell

Expressions used in "assert" should not produce side effects

Analyze your code

Bug Major cert

Since `assert` statements aren't executed by default (they must be enabled with JVM flags) developers should never rely on their execution the evaluation of any logic required for correct program function.

Noncompliant Code Example

```
assert myList.remove(myList.get(0));  // Noncompliant
```

Compliant Solution

```
boolean removed = myList.remove(myList.get(0));
assert removed;
```

See

- CERT, EXP06-J. - Expressions used in assertions must not produce side effects

Available In:

sonarlint sonarcloud sonarqube

Only static class initializers should be used

Code Smell

Empty arrays and collections should be returned instead of null

Code Smell

"@Override" should be used on overriding and implementing methods

Code Smell

Enumeration should not be implemented

Code Smell