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 "for" loop update clause should move the counter in the right direction
Bug

Non-public methods should not be "@Transactional"
Bug

Servlets should not have mutable instance fields
Bug

"toString()" and "clone()" methods should not return null
Bug

".equals()" should not be used to test the values of "Atomic" classes
Bug

Return values from functions without side effects should not be ignored
Bug

Child class methods named for parent class methods should be overrides
Bug

Inappropriate "Collection" calls should not be made
Bug

Silly equality checks should not be made
Bug

Dissimilar primitive wrappers should not be used with the ternary operator without explicit casting
Bug

"InterruptedException" should not be ignored
Bug

Instance methods should not write to "static" fields

Analyze your code

Code Smell Critical multi-threading

Correctly updating a `static` field from a non-static method is tricky to get right and could easily lead to bugs if there are multiple class instances and/or multiple threads in play. Ideally, `static` fields are only updated from `synchronized static` methods.

This rule raises an issue each time a `static` field is updated from a non-static method.

Noncompliant Code Example

```
public class MyClass {

  private static int count = 0;

  public void doSomething() {
    //...
    count++;  // Noncompliant
  }
}
```

Available In:

sonarlint sonarcloud sonarqube

Classes extending java.lang.Thread should override the "run" method

Bug

"Double.longBitsToDouble" should not be used for "int"

Bug

Values should not be uselessly incremented

Bug

Silly String operations should not be made

Bug