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...

name
Bug

Related "if/else if" statements should not have the same condition
Bug

Synchronization should not be done on instances of value-based classes
Bug

"Iterator.hasNext()" should not call "Iterator.next()"
Bug

Identical expressions should not be used on both sides of a binary operator
Bug

Loops with at most one iteration should be refactored
Bug

Variables should not be self-assigned
Bug

"StringBuilder" and "StringBuffer" should not be instantiated with a character
Bug

Methods should not be named "tostring", "hashcode" or "equal"
Bug

"Thread.run()" should not be called directly
Bug

"equals" method overrides should accept "Object" parameters
Bug

The Object.finalize() method should not be called

"for" loop increment clauses should modify the loops' counters

Analyze your code

Code Smell Critical confusing

It can be extremely confusing when a `for` loop's counter is incremented outside of its increment clause. In such cases, the increment should be moved to the loop's increment clause if at all possible.

Noncompliant Code Example

```
for (i = 0; i < 10; j++) { // Noncompliant
  // ...
  i++;
}
```

Compliant Solution

```
for (i = 0; i < 10; i++, j++) {
  // ...
}
```

Or

```
for (i = 0; i < 10; i++) {
  // ...
  j++;
}
```

Available In:

sonarlint | sonarcloud | sonarqube

Bug

Enabling file access for WebViews is security-sensitive

Security Hotspot

Enabling JavaScript support for WebViews is security-sensitive

Security Hotspot

Constructing arguments of system commands from user input is security-sensitive

Security Hotspot

Using unencrypted files in mobile applications is security-sensitive