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...

Code Smell

String operations should not rely on the default system locale

Code Smell

Comments should not be located at the end of lines of code

Code Smell

Track uses of "CHECKSTYLE:OFF" suppression comments

Code Smell

Loggers should be "private static final" and should share a naming convention

Code Smell

Track uses of "NOPMD" suppression comments

Code Smell

Packages should have a javadoc file 'package-info.java'

Code Smell

The members of an interface or class declaration should appear in a pre-defined order

Code Smell

Abstract class names should comply with a naming convention

Code Smell

Strings literals should be placed on the left side when checking for equality

Code Smell

Files should contain an empty newline at the end

Code Smell

Source code should be indented consistently

Synchronized classes Vector, Hashtable, Stack and StringBuffer should not be used

Analyze your code

Code Smell Major performance

Early classes of the Java API, such as `Vector`, `Hashtable` and `StringBuffer`, were synchronized to make them thread-safe. Unfortunately, synchronization has a big negative impact on performance, even when using these collections from a single thread.

It is better to use their new unsynchronized replacements:

- `ArrayList` or `LinkedList` instead of `Vector`
- `Deque` instead of `Stack`
- `HashMap` instead of `Hashtable`
- `StringBuilder` instead of `StringBuffer`

Even when used in synchronized context, you should think twice before using it, since it's usage can be tricky. If you are confident the usage is legitimate, you can safely ignore this warning.

Noncompliant Code Example

```
Vector cats = new Vector();
```

Compliant Solution

```
ArrayList cats = new ArrayList();
```

Exceptions

Use of those synchronized classes is ignored in the signatures of overriding methods.

```
@Override
public Vector getCats() {...}
```

Available In:

sonarlint | sonarcloud | sonarqube

Code Smell

A close curly brace should be located at the beginning of a line

Code Smell

Close curly brace and the next "else", "catch" and "finally" keywords should be on two different lines

Code Smell

Close curly brace and the next "else", "catch" and "finally" keywords should be located on the same line

Code Smell