## 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...

**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**

⚙ 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

**An open curly brace should be located at the beginning of a line**

⚙ Code Smell

**An open curly brace should be located at the end of a line**

⚙ Code Smell

**Tabulation characters should not be used**

⚙ Code Smell

**Functions should not be defined with a variable number of arguments**

⚙ Code Smell

### Anonymous classes should not have too many lines

**Analyze your code**

⚙ Code Smell   ⬢ Major ⓘ

Anonymous classes are a very convenient and compact way to inject a behavior without having to create a dedicated class. But those anonymous inner classes should be used only if the behavior to be injected can be defined in a few lines of code, otherwise the source code can quickly become unreadable.

Available In:

sonarlint | sonarcloud | sonarqube

**Local-Variable Type Inference should be used**

⊗ Code Smell

**Migrate your tests from JUnit4 to the new JUnit5 annotations**

⊗ Code Smell

**Track uses of disallowed classes**

⊗ Code Smell

**Track uses of "@SuppressWarnings" annotations**

⊗ Code Smell