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...

"DateUtils.truncate" from Apache Commons Lang library should not be used
Code Smell

Multiline blocks should be enclosed in curly braces
Code Smell

"readObject" should not be "synchronized"
Code Smell

"Preconditions" and logging arguments should not require evaluation
Code Smell

Boolean expressions should not be gratuitous
Code Smell

"Lock" objects should not be "synchronized"
Code Smell

Classes with only "static" methods should not be instantiated
Code Smell

"Threads" should not be used where "Runnables" are expected
Code Smell

Inner class calls to super class methods should be unambiguous
Code Smell

Unused type parameters should be removed
Code Smell

Parameters should be passed in the correct order
Code Smell

Strings and Boxed types should be compared using "equals()"

Analyze your code

Bug | Major | Quick Fix | cwe cert

It's almost always a mistake to compare two instances of `java.lang.String` or boxed types like `java.lang.Integer` using reference equality `==` or `!=`, because it is not comparing actual value but locations in memory.

Noncompliant Code Example

```
String firstName = getFirstName(); // String overrides equal
String lastName = getLastName();

if (firstName == lastName) { ... }; // Non-compliant; false
```

Compliant Solution

```
String firstName = getFirstName();
String lastName = getLastName();

if (firstName != null && firstName.equals(lastName)) { ... }
```

See

- MITRE, CWE-595 - Comparison of Object References Instead of Object Contents
- MITRE, CWE-597 - Use of Wrong Operator in String Comparison
- CERT, EXP03-J. - Do not use the equality operators when comparing values of boxed primitives
- CERT, EXP50-J. - Do not confuse abstract object equality with reference equality

Available In:

sonarlint | sonarcloud | sonarqube

"ResultSet.isLast()" should not be used

Code Smell

"static" members should be accessed statically

Code Smell

Silly math should not be performed

Code Smell

Classes named like "Exception" should extend "Exception" or a subclass

Code Smell

Exceptions should be either logged or