Security

Hotspot

Tags

(36)

Java static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your JAVA code

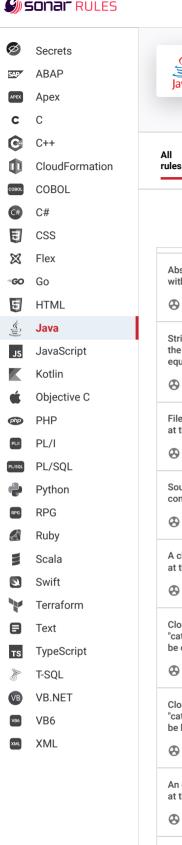
**#** Bug (154)



Products >

(42)

O Quick Fix



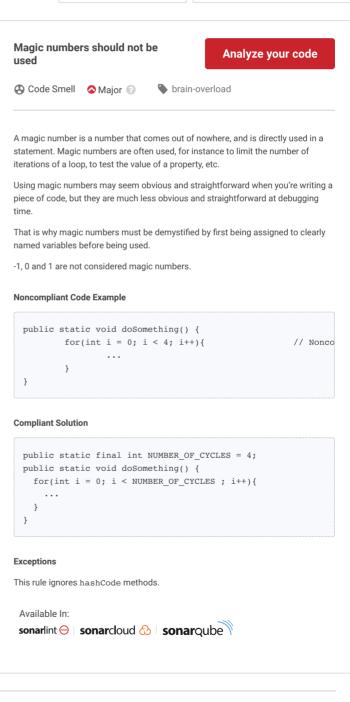
```
Abstract class names should comply
with a naming convention
Code Smell
Strings literals should be placed on
the left side when checking for
equality
Code Smell
Files should contain an empty newline
at the end
A Code Smell
Source code should be indented
consistently
Code Smell
A close curly brace 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

632

6 Vulnerability (53)



⊗ Code

Smell

(389)

Search by name.

© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved. Privacy Policy

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