




 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

Limited dependence should be placed on operator precedence

Analyze your code

Code Smell

Major ?

cwe cert

The rules of operator precedence are complicated and can lead to errors. For this reason, parentheses should be used for clarification in complex statements. However, this does not mean that parentheses should be gratuitously added around every operation.

This rule raises issues when && and || are used in combination, when assignment and equality or relational operators are used in together in a condition, and for other operator combinations according to the following table:

	+ , - , * , / , % < , > , >>	&	^	
+ , - , * , / , %	x	x	x	x
< , > , >>	x	x	x	x
&	x	x	x	x
^	x	x	x	x
	x	x	x	x

This rule also raises an issue when the "true" or "false" expression of a ternary operator is not trivial and not wrapped inside parentheses.

Noncompliant Code Example

```
x = a + b - c;
x = a + 1 << b; // Noncompliant
y = a == b ? a * 2 : a + b; // Noncompliant

if ( a > b || c < d || a == d ) {...}
if ( a > b && c < d || a == b ) {...} // Noncompliant
if ( a = f(b,c) == 1 ) { ... } // Noncompliant; == evaluated f
```

Compliant Solution




```
x = a + b - c;
x = ( a + 1 ) << b;
y = a == b ? ( a * 2 ) : ( a + b );

if ( a > b || c < d || a == d ) {...}
if ( ( a > b && c < d ) || a == b ) {...}
if ( ( a = f(b,c) ) == 1 ) { ... }
```

See

- [CERT, EXP00-C](#) - Use parentheses for precedence of operation
- [CERT, EXP53-J](#) - Use parentheses for precedence of operation
- [MITRE, CWE-783](#) - Operator Precedence Logic Error

Available In:

sonarlint  | sonarcloud  | sonarqube 

https://rules.sonarsource.com/java/RSPEC-864

1/2

<div>Local-Variable Type Inference should be used</div> <div> Code Smell</div>
<div>Migrate your tests from JUnit4 to the new JUnit5 annotations</div> <div> Code Smell</div>
<div>Track uses of disallowed classes</div> <div> Code Smell</div>
<div>Track uses of "@SuppressWarnings" annotations</div> <div> Code Smell</div>

© 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](#)