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

"NullPointerException" should not be caught

Analyze your code

Code Smell | ⊘ Major ⓘ | 🏷 cwe error-handling cert

`NullPointerException` should be avoided, not caught. Any situation in which `NullPointerException` is explicitly caught can easily be converted to a `null` test, and any behavior being carried out in the catch block can easily be moved to the "is null" branch of the conditional.

Noncompliant Code Example

```
public int lengthPlus(String str) {
  int len = 2;
  try {
    len += str.length();
  }
  catch (NullPointerException e) {
    log.info("argument was null");
  }
  return len;
}
```

Compliant Solution

```
public int lengthPlus(String str) {
  int len = 2;

  if (str != null) {
    len += str.length();
  }
  else {
    log.info("argument was null");
  }
  return len;
}
```

See

- MITRE, CWE-395 - Use of NullPointerException Catch to Detect NULL Pointer Dereference
- CERT, ERR08-J. - Do not catch NullPointerException or any of its ancestors

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