

































-  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 rules632

Vulnerability53

Bug154


Security Hotspot36

Code Smell389


Quick Fix42


Tags ▾

Search by name... 🔍


 Code Smell


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


Local-Variable Type Inference should be used




## "Stream" call chains should be simplified when possible

Analyze your code

 Code Smell

 Minor

 clumsy

When using the `Stream` API, call chains should be simplified as much as possible. Not only does it make the code easier to read, it also avoid creating unnecessary temporary objects.

This rule raises an issue when one of the following substitution is possible:

Original	Preferred
<code>stream.filter(predicate).findFirst().isPresent()</code>	<code>stream.anyMatch(predicate)</code>
<code>stream.filter(predicate).findAny().isPresent()</code>	<code>stream.anyMatch(predicate)</code>
<code>!stream.anyMatch(predicate)</code>	<code>stream.noneMatch(predicate)</code>
<code>!stream.anyMatch(x -&gt; !(...))</code>	<code>stream.allMatch(...)</code>
<code>stream.map(mapper).anyMatch(Boolean::booleanValue)</code>	<code>stream.anyMatch(predicate)</code>




### Noncompliant Code Example

```
boolean hasRed = widgets.stream().filter(w -> w.getColor() == RED).findAny().isPresent();
```

### Compliant Solution

```
boolean hasRed = widgets.stream().anyMatch(w -> w.getColor() == RED);
```

Available In:




  

© 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](#)

https://rules.sonarsource.com/java/RSPEC-4034

1/2

<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>