Security

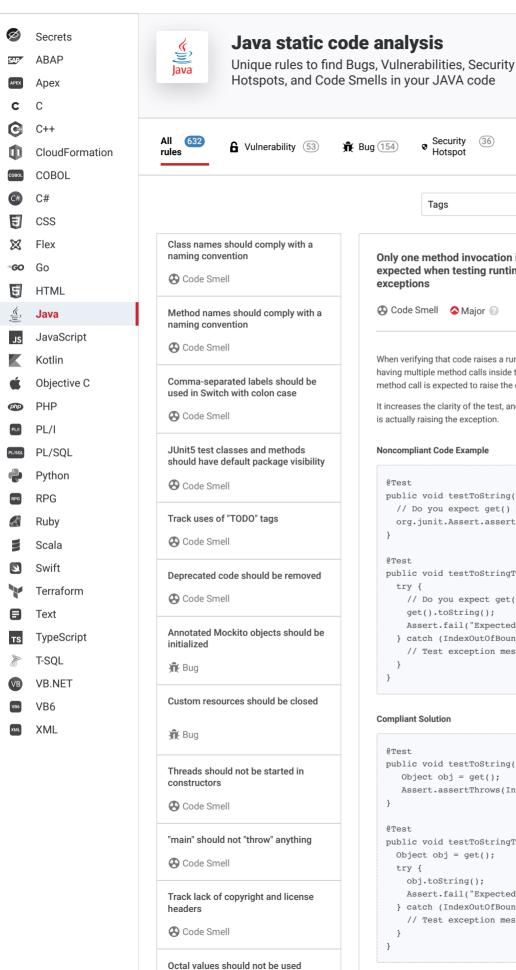
Hotspot

(36)



Products >

O Quick 42 Fix



```
Search by name.
           Tags
Only one method invocation is
expected when testing runtime
                                             Analyze your code
exceptions
Code Smell
               Major 🕝
                               🖣 junit tests
When verifying that code raises a runtime exception, a good practice is to avoid
having multiple method calls inside the tested code, to be explicit about which
method call is expected to raise the exception.
It increases the clarity of the test, and avoid incorrect testing when another method
is actually raising the exception.
Noncompliant Code Example
  @Test
 public void testToString() {
    // Do you expect get() or toString() throwing the exceptio
    org.junit.Assert.assertThrows(IndexOutOfBoundsException.cl
 public void testToStringTryCatchIdiom() {
      // Do you expect get() or toString() throwing the except
      get().toString();
      Assert.fail("Expected an IndexOutOfBoundsException to be
    } catch (IndexOutOfBoundsException e) {
      // Test exception message...
Compliant Solution
  @Test
 public void testToString() {
```

Assert.assertThrows(IndexOutOfBoundsException.class, () -

Assert.fail("Expected an IndexOutOfBoundsException to be

(389)

}

try {

• JUnit exception testing documentation

Object obj = get();

Object obj = get();

obj.toString();

public void testToStringTryCatchIdiom() {

} catch (IndexOutOfBoundsException e) {

// Test exception message...

Code Smell

Exit methods should not be called
HTTP response headers should not be vulnerable to injection attacks  • Vulnerability
Members of Spring components should be injected  • Vulnerability
Classes should not be loaded dynamically  • Vulnerability
Equality operators should not be used

Available In:

sonarlint ⊖ | sonarcloud ↔ | sonarqube

© 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