




 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

bits-1

Bug

"compareTo" should not return "Integer.MIN\_VALUE"

Bug

Boxing and unboxing should not be immediately reversed

Bug

"equals(Object obj)" should test argument type

Bug

"Serializable" inner classes of non-serializable classes should be "static"

Bug

The non-serializable super class of a "Serializable" class should have a no-argument constructor

Bug

Method parameters, caught exceptions and foreach variables' initial values should not be ignored

Bug

"equals(Object obj)" and "hashCode()" should be overridden in pairs

Bug

Disclosing fingerprints from web application technologies is security-sensitive

Security Hotspot

Having a permissive Cross-Origin Resource Sharing policy is security-sensitive

Security Hotspot

Delivering code in production with debug features activated is security-sensitive

Security Hotspot

Inappropriate "Collection" calls should not be made

Analyze your code

BugMajorcert

The java.util.Collection API has methods that accept Object parameters such as Collection.remove(Object o), and Collection.contains(Object o). When the actual type of the object provided to these methods is not consistent with the type declared on the Collection instantiation, those methods will always return false or null. This is most likely unintended and hides a design problem.

This rule raises an issue when the type of the argument of the following APIs is unrelated to the type used for the Collection declaration:

- Collection.remove(Object o)
- Collection.removeAll(Collection<?>)
- Collection.contains(Object o)
- List.indexOf(Object o)
- List.lastIndexOf(Object o)
- Map.containsKey(Object key)
- Map.containsValue(Object value)
- Map.get(Object key)
- Map.getDefault(Object key, V defaultValue)
- Map.remove(Object key)
- Map.remove(Object key, Object value)

Noncompliant Code Example

```
public class S2175 {

    public static void main(String[] args) {
        String foo = "42";
        Map<Integer, Object> map = new HashMap<>();
        map.remove(foo); // Noncompliant; will return 'null' for

        // ...

        List<String> list = new ArrayList<String>();
        Integer integer = Integer.valueOf(1);
        if (list.contains(integer)) { // Noncompliant; always fa
            list.remove(integer); // Noncompliant; list.add(intege
        }
    }
}
```

See

- [CERT, EXP04-J](#). - Do not pass arguments to certain Java Collections Framework methods that are a different type than the collection parameter type

Available In:

sonarlint

sonarcloud

sonarqube


https://rules.sonarsource.com/java/RSPEC-2175

1/2

Searching OS commands in PATH is security-sensitive

 Security Hotspot

Allowing both safe and unsafe HTTP methods is security-sensitive

 Security Hotspot

Creating cookies without the "HttpOnly" flag is security-sensitive

 Security Hotspot

Creating cookies without the "secure" flag is security-sensitive

 Security Hotspot

© 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](#)