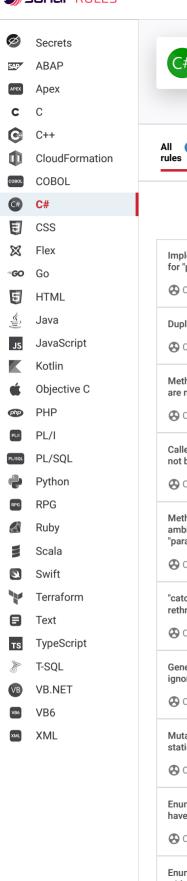
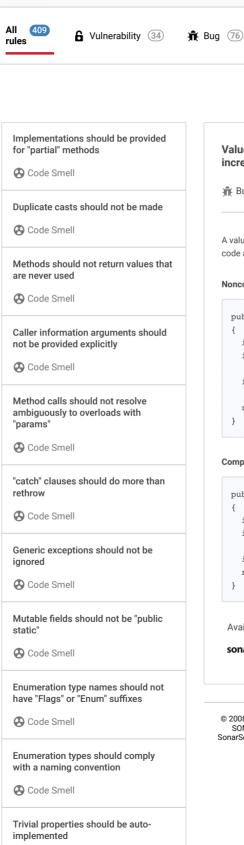
O Quick 52 Fix







Code Smell

simplified

Runtime type checking should be

C# static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your C# code

```
Q
          Tags
                                      Search by name.
Values should not be uselessly
                                            Analyze your code
incremented
👬 Bug 🔷 Major 🕝
                        unused
A value that is incremented or decremented and then not stored is at best wasted
code and at worst a bug.
Noncompliant Code Example
 public int PickNumber()
    int i = 0;
    int j = 0;
    i = i++; // Noncompliant; i is still zero
    return j++; // Noncompliant; 0 returned
Compliant Solution
 public int PickNumber()
    int i = 0;
    int j = 0;
    i++;
    return ++j;
 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

Security

Hotspot

(28)

Boolean checks should not be inverted
Code Smell

Inheritance list should not be redundant
Code Smell

Redundant casts should not be used
Code Smell

Strings should not be concatenated using '+' in a loop