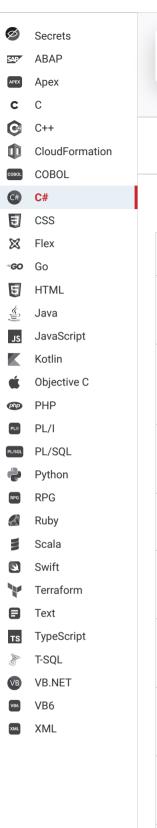
O Quick 52 Fix





```
Hotspots, and Code Smells in your C# code
ΔII
     409
              6 Vulnerability (34)
                                      # Bug (76)
rules
 sealed
 Code Smell
 Overloads with a "StringComparison"
 parameter should be used
 Code Smell
 Overloads with a "CultureInfo" or an
 "IFormatProvider" parameter should
 be used
  Code Smell
 Types should not extend outdated
 base types
 Code Smell
 Properties should be preferred
 Code Smell
 Generics should be used when
 appropriate
 Code Smell
 Type names should not match
 namespaces
 Code Smell
 Strings should be normalized to
 uppercase
 Code Smell
 Exceptions should provide standard
 constructors
 Code Smell
 Assemblies should be marked with
 "NeutralResourcesLanguageAttribute"
 Code Smell
 Interfaces should not be empty
 Code Smell
```

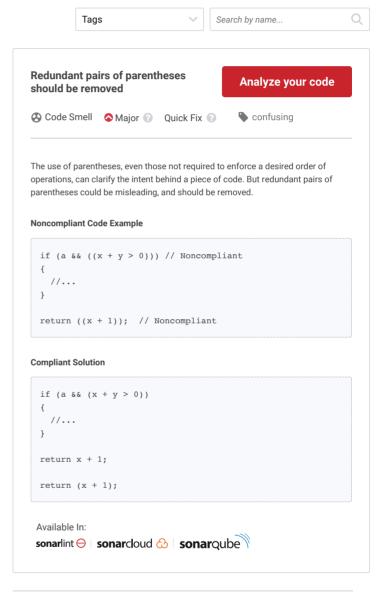
Enumerations should have "Int32"

storage

Code Smell

C# static code analysis

Unique rules to find Bugs, Vulnerabilities, Security



Security

• Hotspot

(28)

© 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

Generic methods should provide type parameters

Code Smell

Multidimensional arrays should not be used

Code Smell

"static readonly" constants should be "const" instead

Code Smell

Strings or integral types should be used for indexers