
































-  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



# C# static code analysis

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

- All rules 409
-  Vulnerability 34
-  Bug 76
-  Security Hotspot 28
-  Code Smell 271
-  Quick Fix 52

Tags ▾

Search by name... 🔍

"protected" members

 Code Smell

Underscores should be used to make large numbers readable

 Code Smell

"ToString()" calls should not be redundant

 Code Smell

"==" should not be used when "Equals" is overridden

 Code Smell

An abstract class should have both abstract and concrete methods

 Code Smell

Multiple variables should not be declared on the same line

 Code Smell

Culture should be specified for "string" operations

 Code Smell

"switch" statements should have at least 3 "case" clauses

 Code Smell

break statements should not be used except for switch cases

 Code Smell

String literals should not be duplicated

 Code Smell

Files should contain an empty newline at the end

 Code Smell

Unused "using" should be removed

 Code Smell

## Assemblies should be marked with "NeutralResourcesLanguageAttribute"

Analyze your code

 Code Smell  Minor  performance

It is important to inform the `ResourceManager` of the language used to display the resources of the neutral culture for an assembly. This improves lookup performance for the first resource loaded.

This rule raises an issue when an assembly contains a `ResX`-based resource but does not have the `System.Resources.NeutralResourcesLanguageAttribute` applied to it.

### Noncompliant Code Example

```
using System;

public class MyClass // Noncompliant
{
    public static void Main()
    {
        string[] cultures = { "de-DE", "en-us", "fr-FR" };
        Random rnd = new Random();
        int index = rnd.Next(0, cultures.Length);
        Thread.CurrentThread.CurrentUICulture = CultureInfo.Cr

        ResourceManager rm = new ResourceManager("MyResources"
                                                    typeof(MyClas
string greeting = rm.GetString("Greeting");

        Console.WriteLine("Enter your name: ");
        string name = Console.ReadLine();
        Console.WriteLine("{0} {1}!", greeting, name);
    }
}
```

### Compliant Solution

```
using System;

[assembly:NeutralResourcesLanguageAttribute("en")]
public class MyClass
{
    public static void Main()
    {
        string[] cultures = { "de-DE", "en-us", "fr-FR" };
        Random rnd = new Random();
        int index = rnd.Next(0, cultures.Length);
        Thread.CurrentThread.CurrentUICulture = CultureInfo.Cr

        ResourceManager rm = new ResourceManager("MyResources"
                                                    typeof(MyClas
string greeting = rm.GetString("Greeting");
```

**A close curly brace should be located at the beginning of a line**

 Code Smell

**Tabulation characters should not be used**

 Code Smell

**Methods and properties should be named in PascalCase**

 Code Smell

**Track uses of in-source issue suppressions**

 Code Smell

```
Console.Write("Enter your name: ");
string name = Console.ReadLine();
Console.WriteLine("{0} {1}!", greeting, name);
}
}
```

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](#)