

- Secrets
- ABAP
- Apex
- C
- C++
- CloudFormation
- COBOL
- C# 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... 🔍

Code Smell

Operator overloads should have named alternatives

Code Smell

Non-abstract attributes should be 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Types should not extend outdated base types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

Local variables should not shadow class fields

Analyze your code

Code Smell Major ? suspicious pitfall





Overriding or shadowing a variable declared in an outer scope can strongly impact the readability, and therefore the maintainability, of a piece of code. Further, it could lead maintainers to introduce bugs because they think they're using one variable but are really using another.

Noncompliant Code Example

```
class Foo
{
    public int myField;

    public void DoSomething()
    {
        int myField = 0; // Noncompliant
        ...
    }
}
```

Available In: sonarlint | sonarcloud | sonarqube

Interfaces should not be empty  Code Smell
Enumerations should have "Int32" storage  Code Smell
Generic methods should provide type parameters  Code Smell
Multidimensional arrays should not be used  Code Smell
"static readonly" constants should be