

- 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...

Code Smell

Native methods should be wrapped

Code Smell

Methods should not have identical implementations

Code Smell

Non-flags enums should not be marked with "FlagsAttribute"

Code Smell

Classes implementing "IEquatable<T>" should be sealed

Code Smell

"GC.SuppressFinalize" should not be called

Code Smell

Objects should not be disposed more than once

Code Smell

Parameter names used into ArgumentException constructors should match an existing one

Code Smell

"ISerializable" should be implemented correctly

Code Smell

"Assembly.Load" should be used

Code Smell

"IDisposable" should be implemented correctly

Code Smell

"ServiceContract" and "OperationContract" attributes should be used together

Code Smell

Flags enumerations zero-value members should be named "None"

Analyze your code

Code Smell Critical convention

Consistent use of "None" in flags enumerations indicates that all flag values are cleared. The value 0 should not be used to indicate any other state, since there is no way to check that the bit 0 is set.

Noncompliant Code Example

```
[Flags]
enum FruitType
{
    Void = 0,           // Non-Compliant
    Banana = 1,
    Orange = 2,
    Strawberry = 4
}
```

Compliant Solution

```
[Flags]
enum FruitType
{
    None = 0,           // Compliant
    Banana = 1,
    Orange = 2,
    Strawberry = 4
}
```

Available In:

sonarlint | sonarcloud | sonarqube

Composite format strings should be used correctly

 Code Smell

Exceptions should not be explicitly rethrown

 Code Smell

"abstract" classes should not have "public" constructors

 Code Smell

Assertion arguments should be passed in the correct order

 Code Smell