

- 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

Child class fields should not differ from parent class fields only by capitalization

Code Smell

Pointers to unmanaged memory should not be visible

Code Smell

Number patterns should be regular

Code Smell

"out" and "ref" parameters should not be used

Code Smell

Unchanged local variables should be "const"

Code Smell

"ConfigureAwait(false)" should be used

Code Smell

"interface" instances should not be cast to concrete types

Code Smell

Literal boolean values should not be used in assertions

Code Smell

Optional parameters should not be used

Code Smell

Public constant members should not be used

Code Smell

Array covariance should not be used

Code Smell

"nameof" should be used

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

Analyze your code

Code Smell Major ?

This rule raises an issue when an externally visible enumeration is marked with `FlagsAttribute` and one, or more, of its values is not a power of 2 or a combination of the other defined values.

Noncompliant Code Example

```
using System;

namespace MyLibrary
{
    [FlagsAttribute]
    public enum Color // Noncompliant, Orange is neither a p
    {
        None      = 0,
        Red        = 1,
        Orange     = 3,
        Yellow     = 4
    }
}
```

Compliant Solution

```
using System;

namespace MyLibrary
{
    public enum Color // Compliant - no FlagsAttribute attri
    {
        None = 0,
        Red = 1,
        Orange = 3,
        Yellow = 4
    }

    [FlagsAttribute]
    public enum Days
    {
        None = 0,
        Monday = 1,
        Tuesday = 2,
        Wednesday = 4,
        Thursday = 8,
        Friday = 16,
        All = Monday | Tuesday | Wednesday | Thursday | Frida
    }
}
```

Available In:



Method: check equality

 Code Smell

Modulus results should not be checked for direct equality

 Code Smell

"for" loop increment clauses should modify the loops' counters

 Code Smell

"switch" statements should not be nested

 Code Smell

Methods and properties should not be too complex