

- 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

Arguments of public methods should be validated against null

Code Smell

Value types should implement "IEquatable<T>"

Code Smell

Finalizers should not be empty

Code Smell

"[ExpectedException]" should not be used

Code Smell

"this" should not be exposed from constructors

Code Smell

Types should not have members with visibility set higher than the type's visibility

Code Smell

Fields should be private

Code Smell

"try" statements with identical "catch" and/or "finally" blocks should be merged

Code Smell

NullReferenceException should not be caught

Code Smell

Functions should not have too many lines of code

Code Smell

"for" loop stop conditions should be invariant

Code Smell

Exceptions should not be thrown from property getters

Analyze your code

Code Smell Major ? error-handling

Property getters should be simple operations that are always safe to call. If exceptions need to be thrown, it is best to convert the property to a method.

It is valid to throw exceptions from indexed property getters and from property setters, which are not detected by this rule.

Noncompliant Code Example

```
public int Foo
{
    get
    {
        throw new Exception(); // Noncompliant
    }
}
```

Compliant Solution

```
public int Foo
{
    get
    {
        return 42;
    }
}
```

Exceptions

No issue is raised when the thrown exception derives from or is of type NotImplementedException, NotSupportedException or InvalidOperationException.

Available In:

sonarlint | sonarcloud | sonarqube

**Statements should be on separate lines**

 Code Smell

**Classes should not be coupled to too many other classes (Single Responsibility Principle)**

 Code Smell

**"switch case" clauses should not have too many lines of code**

 Code Smell

**Magic numbers should not be used**

 Code Smell