- 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

"[ExpectedException]" should not be used

Analyze your code

⊗ Code Smell   🔺 Major ❓   🏷 tests

It should be clear to a casual reader what code a test is testing and what results are expected. Unfortunately, that's not usually the case with the `ExpectedException` attribute since an exception could be thrown from almost any line in the method.

This rule detects MSTest and NUnit `ExpectedException` attribute.

Noncompliant Code Example

```
[TestMethod]
[ExpectedException(typeof(ArgumentNullException))]  // Nonco
public void TestNullArg()
{
  //...
}
```

Compliant Solution

```
[TestMethod]
public void TestNullArg()
{
  bool callFailed = false;
  try
  {
    //...
  }
  catch (ArgumentNullException)
  {
    callFailed = true;
  }
  Assert.IsTrue(callFailed, "Expected call to MyMethod to fa
}
```

or

```
[TestMethod]
public void TestNullArg()
{
  Assert.ThrowsException<ArgumentNullException>(() => /*...*
}
```

Exceptions

This rule ignores one-line test methods, since it is obvious in such methods where the exception is expected to be thrown.

Available In:

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

sonarlint ⊙ | sonarcloud ⊙ | sonarqube))