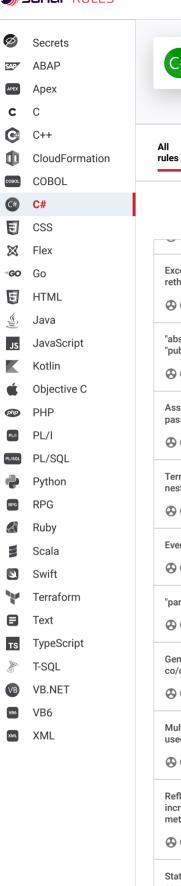
Analyze your code

pitfall





```
C# static code analysis
Unique rules to find Bugs, Vulnerabilities, Security
Hotspots, and Code Smells in your C# code

All 409
rules

All 40
```

Method overrides should not

change parameter defaults



Fields that are only assigned in the

## Default arguments are determined by the static type of the object. If a default argument is different for a parameter in an overriding method, the value used in the call will be different when calls are made via the base or derived object, which may be contrary to developer expectations. Default parameter values are useless in explicit interface implementations, because the static type of the object will always be the implemented interface. Thus, specifying default values is useless and confusing. **Noncompliant Code Example** using System; public class Base public virtual void Write(int i = 42) Console.WriteLine(i); } public class Derived : Base public override void Write(int i = 5) // Noncompliant Console.WriteLine(i); public class Program public static void Main() var derived = new Derived(); derived.Write(); // writes 5 Print(derived); // writes 42; was that expected? private static void Print(Base item) item.Write(); **Compliant Solution**

```
using System;
public class Base
```

сопѕитистог ѕпоита ре теацопту

**⇔** Code Smell

Static fields should not be used in generic types

Code Smell

Multiline blocks should be enclosed in curly braces

Code Smell

Boolean expressions should not be gratuitous

Code Smell

Types and methods should not have

```
public virtual void Write(int i = 42)
        Console.WriteLine(i);
public class Derived : Base
   public override void Write(int i = 42)
        Console.WriteLine(i);
public class Program
   public static void Main()
        var derived = new Derived();
       derived.Write(); // writes 42
       Print(derived); // writes 42
   private static void Print(Base item)
        item.Write();
}
```

Available In:

sonarlint ⊕ | sonarcloud & | sonarqube

© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved.

Privacy Policy