# Lab: Style Cop

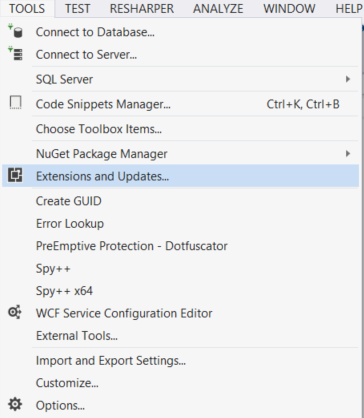
This document defines the lab overview for the ["C# OOP Advanced" course @ Software University](https://softuni.bg/c-sharp-advanced-oop). Please submit your solutions (source code) of all below described problems at the end of the course at [softuni.bg](https://softuni.bg/trainings/1376/java-advanced-oop-july-2016).

# Introduction

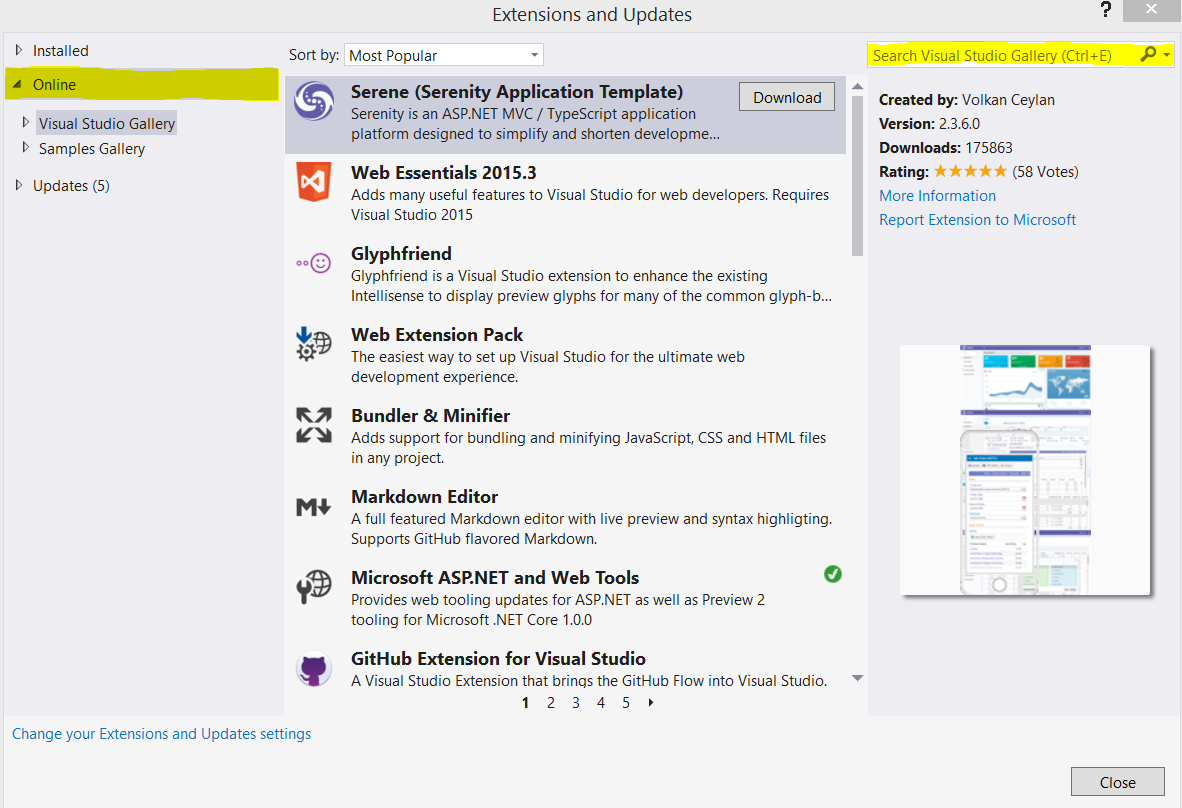
Since we are learning how to write **readable** and “**beautiful**” code, by following the code conventions, it is sometimes hard to remember all those strange rules, like for example that we need to put an empty line after each method. Luckily, some good people have made tools that help us find such “errors” and tell us how to fix them. One of those tools is called Style Cop and it is quite cool actually. Most importantly, it’s easy to use.

## Installing Style Cop

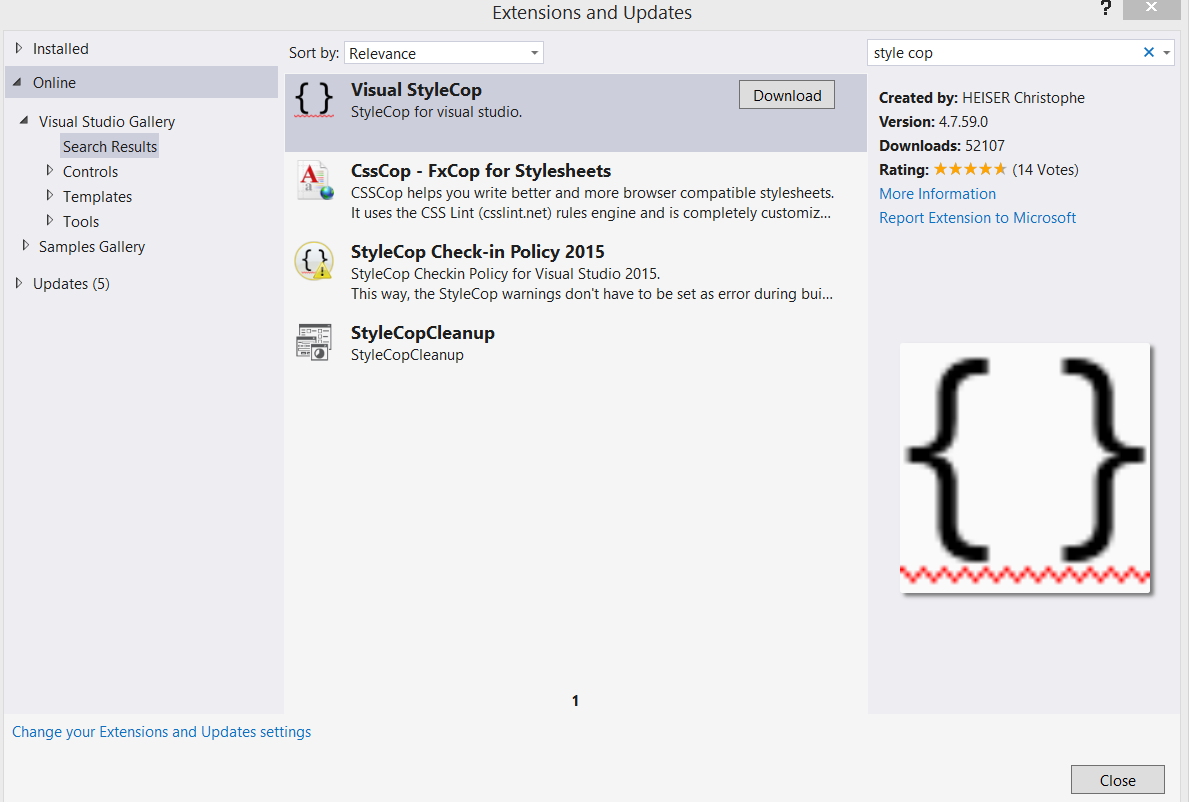
Once we’ve opened a project, we can go to Tools and select Extensions and Updates..



After we’ve clicked that and a window has opened, we need to click the Online category.

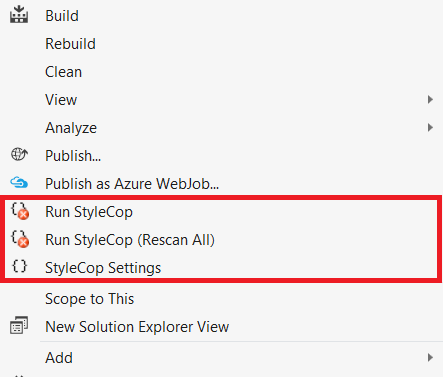


Here we have to search for style cop and the first result should be the following:



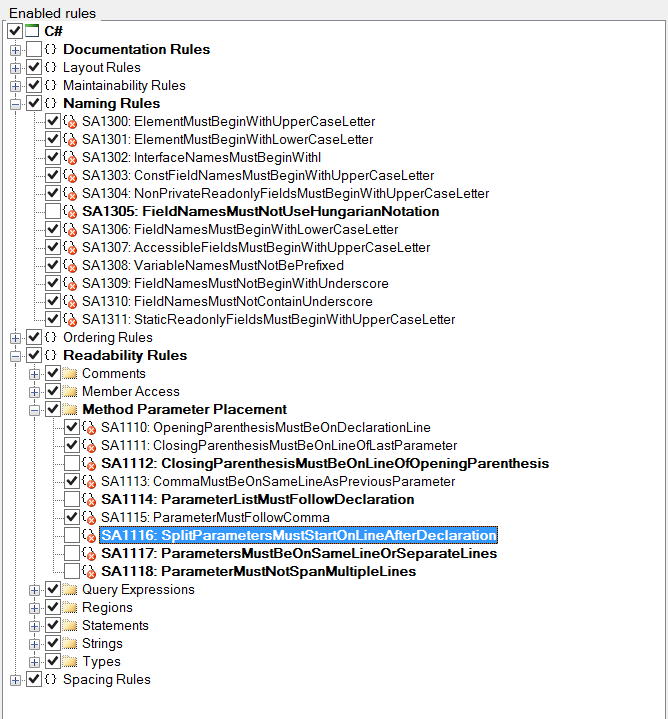
## Configuring Style Cop

After **downloading** it and **installing** it and **restarting** visual studio, it should be integrated in your environment. So now after right clicking on the project in the solution explorer you should see 3 new buttons.



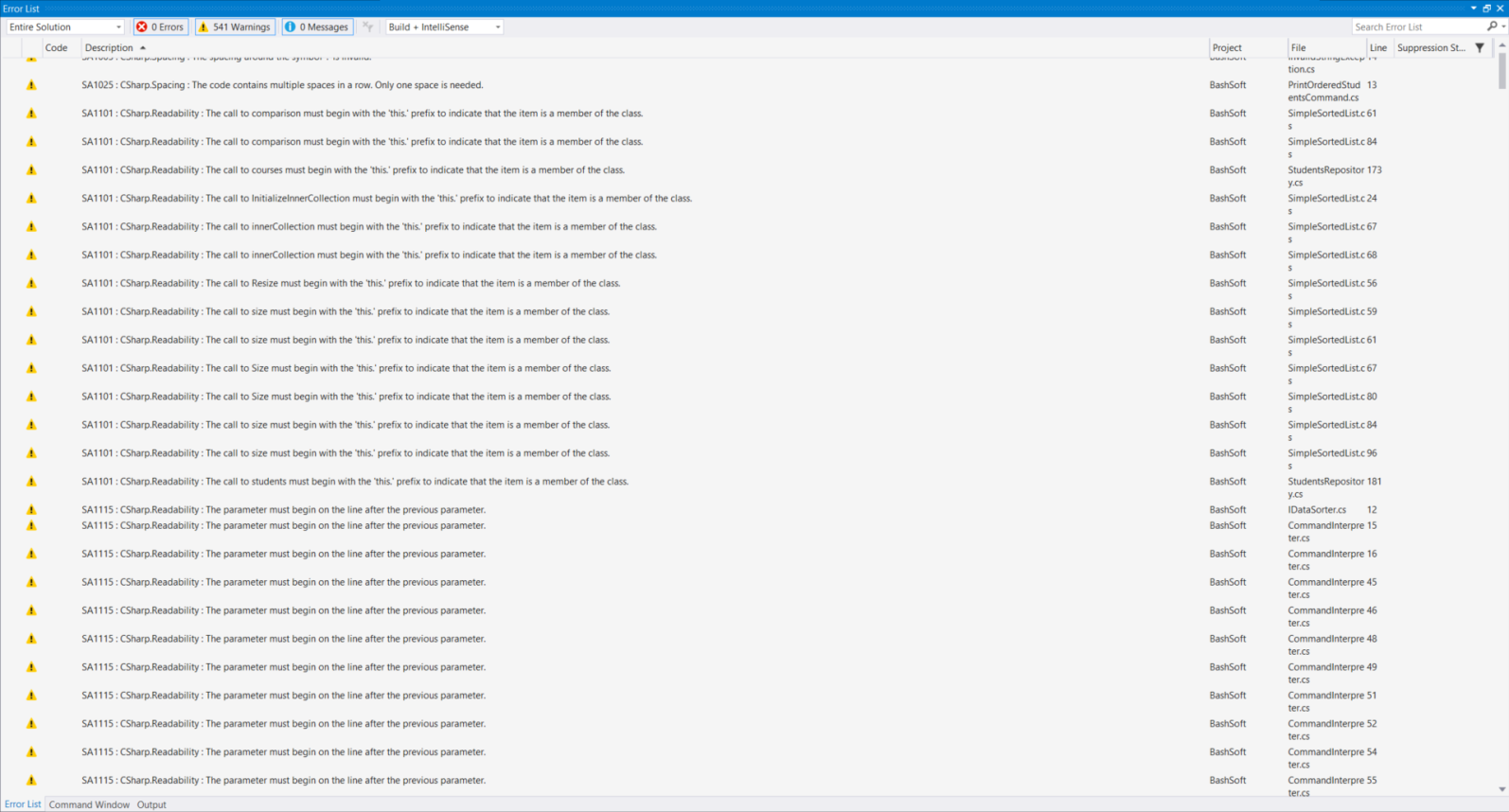
Before reading it, we should turn off some things in the settings. If you don’t, it will automatically warn you that you do not have documentation added above each class/interface/method and so on, and that is something we do not want for the current project. So you need to click **StyleCop** **Settings** -> **check** **out** the **Documentation** **Rules**. There are also a few other things, we might want to exclude for checks in our project. Those are the warnings that will appear if you put the signature of a **method**/**constructor** on two lines. We’ve made the signatures on more than one line, because not everybody has a 4K 30’ displays and that way it's more readable for an average programmer. So the things you might want to disable are in the **readability rules** -> **method parameter placement -> SA1112, SA1114, SA1116, SA1117, SA 1118.** Finally, the last thing we need to uncheck is in the **Naming rules -> SA1305.**

After that the window should look like this:



Now you should click on the project and Run StyleCop.

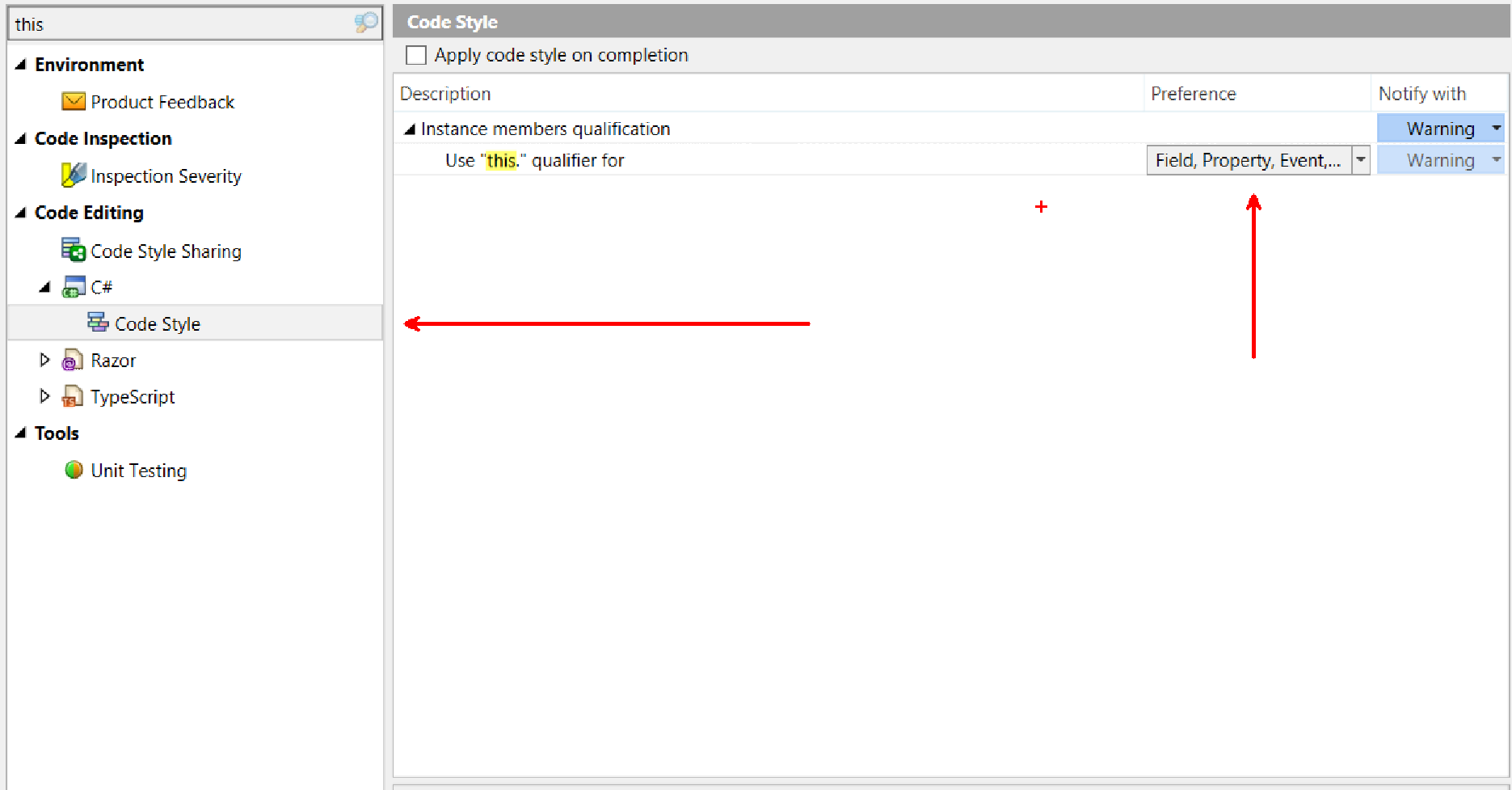
There should be around 600 warnings which you might fix, in order to be following all the conventions.



## Additional setting for ReSharper

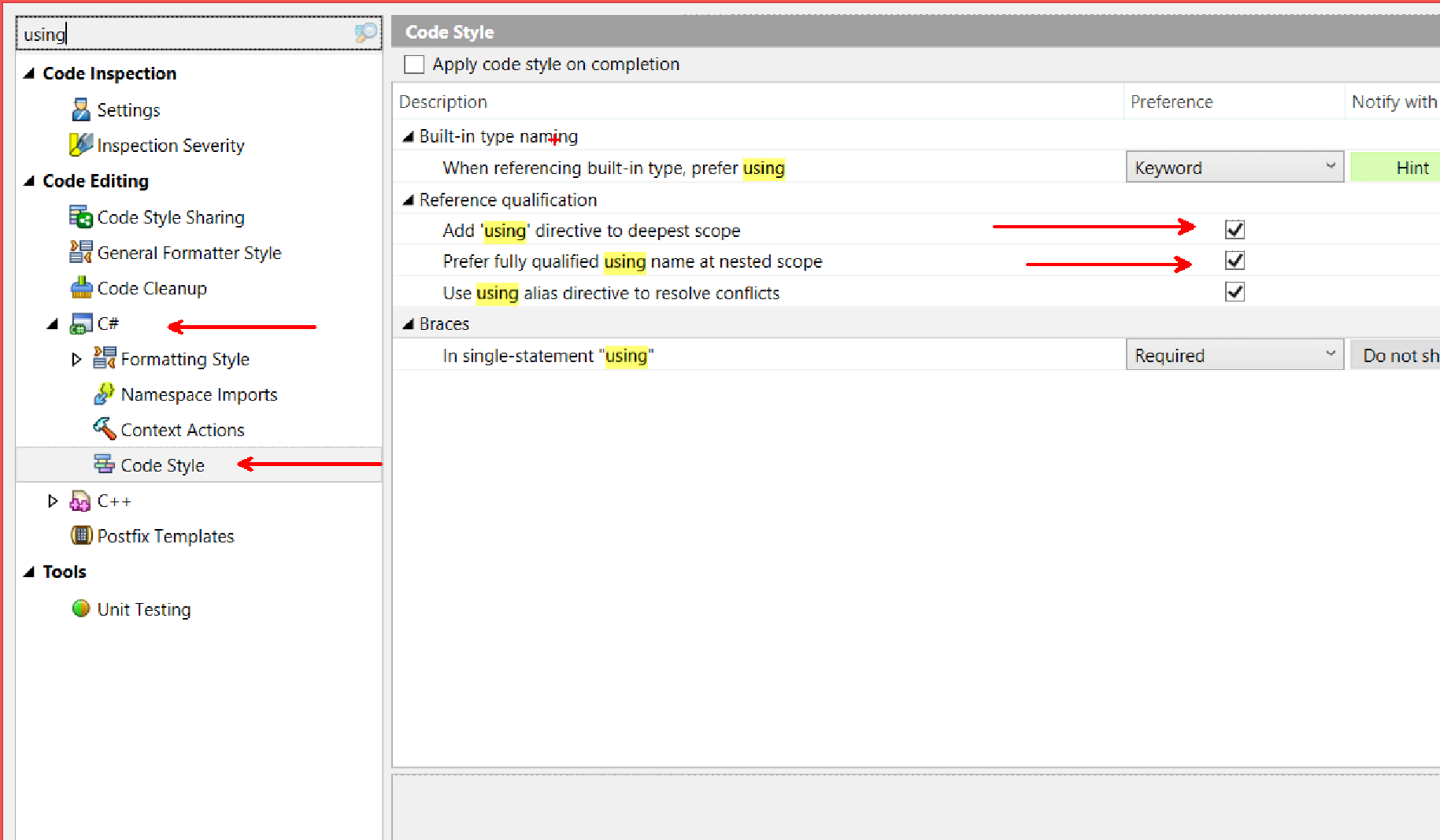
There are some things that are in conflicts between the **Re#er** conventions and the **StyleCop** conventions. One of them is the **using** of “**this**” when calling something from the current object.

Go to Resharper in the main tab -> Options and type in “this” in the search in the upper left corner of the window.



As you can see you should open Code style in C# and change the preference of the usage of “this” to all the possible class members (field, property, event, method). This way **re#er** will add “this” automatically if you want to use a non-static member in the current class.

Another thing that you will notice is that style cop tells us that the using statements need to be inside the namespace. That’s why you might want to check one more thing in the **Re#er.** Open the Options and type “using” in the search. And result in the following configuration:



Hopefully JetBrains will soon make a good IDE for C# like IntelliJ is for Java and we will have all those conventions implemented automatically in the IDE and not using a tool that makes Visual Studio even slower.