# Contributing

If you want to contribute code you can get started by looking for issues marked as

[up for grabs](https://github.com/DotNetAnalyzers/StyleCopAnalyzers/labels/up%20for%20grabs).

We also have the [easy](https://github.com/DotNetAnalyzers/StyleCopAnalyzers/labels/easy) tag

for issues suitable if you are unfamiliar with roslyn.

You can also help by filing issues, participating in discussions and doing code review.

## Building prerequisites

\* Visual Studio 2017 (Community Edition or higher) is required for building this repository.

\* The version of the [.NET Core SDK](https://dotnet.microsoft.com/download/dotnet-core) as specified in the global.json file at the root of this repo.

Use the init script at the root of the repo to conveniently acquire and install the right version.

## Implementing a diagnostic

1. To start working on a diagnostic, add a comment to the issue indicating you are working on implementing it.

2. Add a new issue for a code fix for the diagnostic. For example, I added #171 when I worked on #6. Even if no code fix

is possible, the issue is a place for discussions regarding possible corrections. Code fixes may, but do not have to

be implemented alongside the diagnostic.

3. If a diagnostic or code fix is submitted without tests, it might be rejected. However, it may be accepted provided

all of the following are true:

1. The code is disabled by default, by passing `AnalyzerConstants.DisabledNoTests` for the `isEnabledByDefault`

parameter when creating the `DiagnosticDescriptor`. It will be enabled by default only after tests are in place.

2. A new issue was created for implementing tests for the item (e.g. #176).

3. Evidence was given that the feature is currently operational, and the code appears to be a solid starting point

for other contributors to continue the implementation effort.