# How to become a contributor and submit your own code

## Contributor License Agreements

We'd love to accept your patches! Before we can take them, we have to jump a

couple of legal hurdles.

Please fill out either the individual or corporate Contributor License Agreement

(CLA).

\* If you are an individual writing original source code and you're sure you

own the intellectual property, then you'll need to sign an

[individual CLA](https://developers.google.com/open-source/cla/individual).

\* If you work for a company that wants to allow you to contribute your work,

then you'll need to sign a

[corporate CLA](https://developers.google.com/open-source/cla/corporate).

Follow either of the two links above to access the appropriate CLA and

instructions for how to sign and return it. Once we receive it, we'll be able to

accept your pull requests.

## Are you a Googler?

If you are a Googler, please make an attempt to submit an internal change rather

than a GitHub Pull Request. If you are not able to submit an internal change a

PR is acceptable as an alternative.

## Contributing A Patch

1. Submit an issue describing your proposed change to the

[issue tracker](https://github.com/google/googletest/issues).

2. Please don't mix more than one logical change per submittal, because it

makes the history hard to follow. If you want to make a change that doesn't

have a corresponding issue in the issue tracker, please create one.

3. Also, coordinate with team members that are listed on the issue in question.

This ensures that work isn't being duplicated and communicating your plan

early also generally leads to better patches.

4. If your proposed change is accepted, and you haven't already done so, sign a

Contributor License Agreement (see details above).

5. Fork the desired repo, develop and test your code changes.

6. Ensure that your code adheres to the existing style in the sample to which

you are contributing.

7. Ensure that your code has an appropriate set of unit tests which all pass.

8. Submit a pull request.

## The Google Test and Google Mock Communities

The Google Test community exists primarily through the

[discussion group](http://groups.google.com/group/googletestframework) and the

GitHub repository. Likewise, the Google Mock community exists primarily through

their own [discussion group](http://groups.google.com/group/googlemock). You are

definitely encouraged to contribute to the discussion and you can also help us

to keep the effectiveness of the group high by following and promoting the

guidelines listed here.

### Please Be Friendly

Showing courtesy and respect to others is a vital part of the Google culture,

and we strongly encourage everyone participating in Google Test development to

join us in accepting nothing less. Of course, being courteous is not the same as

failing to constructively disagree with each other, but it does mean that we

should be respectful of each other when enumerating the 42 technical reasons

that a particular proposal may not be the best choice. There's never a reason to

be antagonistic or dismissive toward anyone who is sincerely trying to

contribute to a discussion.

Sure, C++ testing is serious business and all that, but it's also a lot of fun.

Let's keep it that way. Let's strive to be one of the friendliest communities in

all of open source.

As always, discuss Google Test in the official GoogleTest discussion group. You

don't have to actually submit code in order to sign up. Your participation

itself is a valuable contribution.

## Style

To keep the source consistent, readable, diffable and easy to merge, we use a

fairly rigid coding style, as defined by the

[google-styleguide](https://github.com/google/styleguide) project. All patches

will be expected to conform to the style outlined

[here](https://google.github.io/styleguide/cppguide.html). Use

[.clang-format](https://github.com/google/googletest/blob/master/.clang-format)

to check your formatting.

## Requirements for Contributors

If you plan to contribute a patch, you need to build Google Test, Google Mock,

and their own tests from a git checkout, which has further requirements:

\* [Python](https://www.python.org/) v2.3 or newer (for running some of the

tests and re-generating certain source files from templates)

\* [CMake](https://cmake.org/) v2.6.4 or newer

## Developing Google Test and Google Mock

This section discusses how to make your own changes to the Google Test project.

### Testing Google Test and Google Mock Themselves

To make sure your changes work as intended and don't break existing

functionality, you'll want to compile and run Google Test and GoogleMock's own

tests. For that you can use CMake:

mkdir mybuild

cd mybuild

cmake -Dgtest\_build\_tests=ON -Dgmock\_build\_tests=ON ${GTEST\_REPO\_DIR}

To choose between building only Google Test or Google Mock, you may modify your

cmake command to be one of each

cmake -Dgtest\_build\_tests=ON ${GTEST\_DIR} # sets up Google Test tests

cmake -Dgmock\_build\_tests=ON ${GMOCK\_DIR} # sets up Google Mock tests

Make sure you have Python installed, as some of Google Test's tests are written

in Python. If the cmake command complains about not being able to find Python

(`Could NOT find PythonInterp (missing: PYTHON\_EXECUTABLE)`), try telling it

explicitly where your Python executable can be found:

cmake -DPYTHON\_EXECUTABLE=path/to/python ...

Next, you can build Google Test and / or Google Mock and all desired tests. On

\\*nix, this is usually done by

make

To run the tests, do

make test

All tests should pass.

### Regenerating Source Files

Some of Google Test's source files are generated from templates (not in the C++

sense) using a script. For example, the file

\*googlemock/include/gmock/gmock-generated-actions.h.pump\* is used to generate

\*gmock-generated-actions.h\* in the same directory.

You don't need to worry about regenerating the source files unless you need to

modify them. You would then modify the corresponding `.pump` files and run the

'[pump.py](googlemock/scripts/pump.py)' generator script. See the

[Pump Manual](googlemock/docs/pump\_manual.md).