# How to become a contributor and submit your own code

## Contributor License Agreements

We'd love to accept your sample apps and 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.

## Contributing A Patch

1. Submit an issue describing your proposed change to the repo in question.

1. The repo owner will respond to your issue promptly.

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

Contributor License Agreement (see details above).

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

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

you are contributing.

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

1. Submit a pull request.

## Setting up a development environment

\* [Mac development environment guide](MAC\_SETUP.md)

## Authoring, testing, and contributing samples

See [AUTHORING\_GUIDE.md](AUTHORING\_GUIDE.md).

## Code Reviews

After meeting the above criteria, your code will need to be approved by two reviewers before it can be merged into master. One will be a [CODEOWNER](.github/CODEOWNERS) for the product you are contributing to, and the other will be a repo owner, there to double check for anything that might be detrimental to the overall repo health (things that could cause future tech debt, test flakiness, etc.). Both will automatically be assigned. Some product areas have mulitple folks who can act as the CODEOWNER, and you may be working more closely with a teammate who isn't the automatically assigned reviewer. In that case, it is perfectly fine to manually assign the teammate more familiar with this work as your CODEOWNER reviewer. If you do not hear from your repo owner reviewer within a day (and you know they are not OOO), send them a friendly ping so that you can better understand the review cadence for your PR. All the repo owners are juggling reviews alongside other work, and their velocities can vary, but they are happy to hear from you. If you see that your repo owner reviewer is OOO, you can use the "blunderbuss: assign" label to assign a new reviewer.