How to contribute

=================

Thank you so much for wanting to contribute to Guava! Here are a few important

things you should know about contributing:

1. API changes require discussion, use cases, etc. Code comes later.

2. Pull requests are great for small fixes for bugs, documentation, etc.

3. Pull requests are not merged directly into the master branch.

3. Code contributions require signing a Google CLA.

API changes

-----------

We make changes to Guava's public [APIs][], including adding new APIs, very

carefully. Because of this, if you're interested in seeing a new feature in

Guava, the best approach is to create an [issue][] (or comment on an existing

issue if there is one) requesting the feature and describing specific use cases

for it.

If the feature has merit, it will go through a thorough process of API design

and review. Any code should come after this.

[APIs]: http://en.wikipedia.org/wiki/Application\_programming\_interface

[issue]: https://github.com/google/guava/issues

Pull requests

-------------

Unless the change is a trivial fix such as for a typo, it's generally best to

start by opening a new issue describing the bug or feature you're intending to

fix. Even if you think it's relatively minor, it's helpful to know what people

are working on. And as mentioned above, API changes should be discussed

thoroughly before moving to code.

Some examples of types of pull requests that are immediately helpful:

- Fixing a bug without changing a public API.

- Fixing or improving documentation.

- Improvements to Maven configuration.

Guidelines for any code contributions:

1. Any significant changes should be accompanied by tests. The project already

has good test coverage, so look at some of the existing tests if you're

unsure how to go about it.

2. All contributions must be licensed Apache 2.0 and all files must have a

copy of the boilerplate license comment (can be copied from an existing

file).

3. Files should be formatted according to Google's [Java style guide][].

4. Please squash all commits for a change into a single commit (this can be

done using `git rebase -i`). Do your best to have a

[well-formed commit message][] for the change.

[Java style guide]: https://google.github.io/styleguide/javaguide.html

[well-formed commit message]: http://tbaggery.com/2008/04/19/a-note-about-git-commit-messages.html

#### Merging pull requests ####

Due to Guava's nature as a subset of Google's internal codebase which is

automatically synced to the public GitHub repository, we are unable to merge

pull requests directly into the master branch. Instead, once a pull request is

ready for merging, we'll make the appropriate changes in the internal codebase

and, when the change is synced out, give the pull request author credit for the

commit.

Contributor License Agreement

-----------------------------

Contributions to any Google project must be accompanied by a Contributor

License Agreement. This is not a copyright \_assignment\_; it simply gives

Google permission to use and redistribute your contributions as part of the

project.

- 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][]. Please include your GitHub username.

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

then you'll need to sign a [corporate CLA][].

You generally only need to submit a CLA once, so if you've already submitted

one (even if it was for a different project), you probably don't need to do it

again.

[individual CLA]: https://cla.developers.google.com/about/google-individual

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