# Contributing

Spring Cloud is released under the non-restrictive Apache 2.0 license,

and follows a very standard Github development process, using Github

tracker for issues and merging pull requests into master. If you want

to contribute even something trivial please do not hesitate, but

follow the guidelines below.

## Sign the Contributor License Agreement

Before we accept a non-trivial patch or pull request we will need you to sign the

[Contributor License Agreement](https://cla.pivotal.io/sign/spring).

Signing the contributor's agreement does not grant anyone commit rights to the main

repository, but it does mean that we can accept your contributions, and you will get an

author credit if we do. Active contributors might be asked to join the core team, and

given the ability to merge pull requests.

## Code of Conduct

This project adheres to the Contributor Covenant [Code of conduct](https://github.com/spring-cloud/spring-cloud-build/blob/master/docs/src/main/asciidoc/code-of-conduct.adoc). By participating, you are expected to uphold this code. Please report

unacceptable behavior to spring-code-of-conduct@pivotal.io.

## Code Conventions and Housekeeping

None of these is essential for a pull request, but they will all help. They can also be

added after the original pull request but before a merge.

\* Use the Spring Framework code format conventions. If you use Eclipse

you can import formatter settings using the

`eclipse-code-formatter.xml` file from the

[Spring Cloud Build](https://raw.githubusercontent.com/spring-cloud/spring-cloud-build/master/spring-cloud-dependencies-parent/eclipse-code-formatter.xml) project. If using IntelliJ, you can use the

[Eclipse Code Formatter Plugin](https://plugins.jetbrains.com/plugin/6546) to import the same file.

\* Make sure all new `.java` files to have a simple Javadoc class comment with at least an

`@author` tag identifying you, and preferably at least a paragraph on what the class is

for.

\* Add the ASF license header comment to all new `.java` files (copy from existing files

in the project)

\* Add yourself as an `@author` to the .java files that you modify substantially (more

than cosmetic changes).

\* Add some Javadocs and, if you change the namespace, some XSD doc elements.

\* A few unit tests would help a lot as well -- someone has to do it.

\* If no-one else is using your branch, please rebase it against the current master (or

other target branch in the main project).

\* When writing a commit message please follow [these conventions](https://tbaggery.com/2008/04/19/a-note-about-git-commit-messages.html),

if you are fixing an existing issue please add `Fixes gh-XXXX` at the end of the commit

message (where XXXX is the issue number).