# Contributing

Contributions to Foundation are welcome! This project follows the [contribution guidelines for the Swift project](https://swift.org/contributing/#contributing-code). If you are interested in contributing, please consult with our [status page](Docs/Status.md) to see what work remains to be done. A few additional details are outlined below.

## Licensing

By submitting a pull request, you represent that you have the right to license your contribution to Apple and the community, and agree by submitting the patch that your contributions are licensed under the [Swift license](https://swift.org/LICENSE.txt).

## Bug Reports

You can use the `Foundation` component in the [bug reporter](https://bugs.swift.org) if you know your bug is specifically about Swift Foundation.

Please remember to include platform information with your report. If the bug is about the Foundation framework on Darwin, then please use [Feedback Assistant](https://feedbackassistant.apple.com).

## Pull Requests

Before embarking on a large amount of work to implement missing functionality, please double-check with the community on the [swift-corelibs-dev](https://lists.swift.org/mailman/listinfo/swift-corelibs-dev) mailing list. Someone may already be working in this area, and we want to avoid duplication of work.

If your request includes functionality changes, please be sure to test your code on Linux as well as macOS. Differences in the compiler and runtime on each platform means that code that compiles and runs correctly on Darwin (where the Objective-C runtime is present) may not compile at all on Linux.

##### Review

Each pull request will be reviewed by a code owner before merging.

\* Pull requests should contain small, incremental change.

\* Focus on one task. If a pull request contains several unrelated commits, we will ask for the pull request to be split up.

\* Please squash work-in-progress commits. Each commit should stand on its own (including the addition of tests if possible). This allows us to bisect issues more effectively.

\* After addressing review feedback, please rebase your commit so that we create a clean history in the `master` branch.

##### Tests

All pull requests which contain code changes should come with a new set of automated tests, and every current test must pass on all supported platforms.

##### Documentation

Most of the methods in Foundation are lacking documentation. We appreciate your help in filling out documentation when you implement a method. Use the markdown syntax in the [Swift Naming Guidelines](https://swift.org/documentation/api-design-guidelines.html#write-doc-comment).

##### Implementation Status

We are currently keeping track of the implementation status and test coverage status of our public classes on our [status page](Docs/Status.md). If you make changes to our public interfaces, please take care to update the corresponding entries in that document.

## API Changes

The interface of Foundation is intended to be both stable and cross-platform. This means that when API is added to Foundation, it is effectively permanent.

It is therefore critical that any code change that affects the public-facing API go through a full `swift-evolution` review process. This gives us the chance to ensure several important requirements are satisfied:

\* The proposal aligns with our current goals for the upcoming release.

\* We are comfortable supporting the proposed API for the long term.

\* We believe we can make the same change to the API of Darwin Foundation. This could be done via changes in the overlay, changes in the compiler, or changes in Darwin Foundation itself. This must be addressed in every proposal.