# Contributing to Mocha

> Please read these guidelines before submitting an issue, filing a feature request, or contributing code.

## :question: Got a Question?

If you have a question about using Mocha, please use the [mailing list](https://groups.google.com/group/mochajs), [StackOverflow](https://stackoverflow.com), or ask the friendly people in our [chat room](https://gitter.im/mochajs/mocha).

## :bug: I Found a Bug

Sorry! It happens to the best of us. If you've found a bug in Mocha, \*\*please [search](https://github.com/mochajs/mocha/issues/) to see if it's already been reported\*\*. Otherwise, create a [new issue](https://github.com/mochajs/mocha/issues/new). If you can fix the bug yourself, feel free to create a [pull request](#propose-a-change) thereafter.

Please include \_as much detail as possible\_ to help us reproduce and diagnose the bug. Most importantly:

- Let us know \_how\_ you're running Mocha (options, flags, environment, browser or Node.js, etc.)

- Include your test code or file(s). If large, please provide a link to a repository or [gist](https://gist.github.com).

- Please show code in JavaScript only (any version)

If we need more information from you, we'll let you know. If you don't within a reasonable time frame (TBD), your issue will be automatically closed for inactivity.

## :exclamation: Propose a Change

Before you get your hands dirty, please [search](https://github.com/mochajs/mocha/issues/) for a related issue, or [create a new one](https://github.com/mochajs/mocha/issues/new). If you wish to contribute a new feature, this is doubly important! Let's discuss your proposed changes first; we don't want you to waste time implementing a change that is at odds with the project's direction. That said, we'll happily consider any contribution, no matter how great or small.

\_This paragraph would contain information about Mocha's roadmap, but it doesn't yet exist.\_ :poop:

It's also important to understand some overarching goals of Mocha, detailed below.

### :soccer: About Project Goals

Mocha is a test framework. Developers use it against anything from legacy spaghetti in IE11 to stage-0 TC39 features in Electron. Mocha is committed to providing support for maintained (LTS) versions of Node.js and popular browsers (of which IE11 is still one, as of December 2018).

Mocha adheres strictly to [semantic versioning](https://semver.org). We are \_extremely cautious\_ with changes that have the potential to break; given the size of Mocha's user base, it's \_highly unlikely\_ a breaking change will slide by.

Mocha's usage far outweighs its resources. If a proposed feature would incur a maintenance penalty, it could be a hard sell.

We ask you please keep these goals in mind when making or proposing changes.

### :shoe: Contributing Code: Step-by-Step

Follow these steps to get going. If you are having trouble, don't be afraid to [ask for help](#got-a-question).

> PRO TIP: After `npm install`, run `npm start` to see a list of commands which can be run with `npm start <command>` (powered by [nps](https://npm.im/nps)).

1. [Install Node.js 10.12.0 or newer](https://nodejs.org/en/download/).

- If you're new to installing Node, a tool like [nvm](https://github.com/creationix/nvm#install-script) can help you manage multiple version installations.

- You will need [Google Chrome](https://www.google.com/chrome/) to run browser-based tests locally.

1. Follow [Github's documentation](https://help.github.com/articles/fork-a-repo/) on setting up Git, forking and cloning.

1. Create a new branch in your working copy. Give your branch a descriptive name, such as `issue/12345`: `git checkout -b issue/12345`.

1. Execute `npm install` to install the development dependencies.

- Do not use `yarn install`.

- Some optional dependencies may fail; you can safely ignore these unless you are trying to build the documentation.

- If you're sick of seeing the failures, run `npm install --ignore-scripts`.

1. Make your changes and add them via `git add`.

- Your changes will likely be somewhere in `lib/`, `bin/` or `browser-entry.js` (if your changes are browser-specific).

- Unit and/or integration \*\*tests are required\*\* for any code change. These live in `test/`.

- \*\*Do not modify\*\* the root `mocha.js` file directly; it is automatically generated.

- Keep your PR focused. Don't fix two things at once; don't upgrade dependencies unless necessary.

1. Before committing, run `npm start test`.

- This will run both Node.js-based and browser-based tests.

- Ultimately, your pull request will be built on our continuous integration servers ([Travis CI](https://travis-ci.org/mochajs/mocha) and [AppVeyor](https://ci.appveyor.com/project/boneskull/mocha)). The first step to ensuring these checks pass is to test on your own machine.

- A coverage check will be sent to [Coveralls](https://coveralls.io/github/mochajs/mocha). \*\*A drop in code coverage % is considered a failed check\*\*.

1. Commit your changes.

- Use a brief message on the first line, referencing a relevant issue (e.g. `closes #12345`).

- Add detail in subsequent lines.

- A pre-commit hook will run which automatically formats your staged changes (and fixes any problems it can) with ESLint and Prettier. If ESLint fails to fix an issue, your commit will fail and you will need to manually correct the problem.

1. <a name="up-to-date"/> (Optional) Ensure you are up-to-date with Mocha's `master` branch:

- You can add an "upstream" remote repo using `git remote add upstream https://github.com/mochajs/mocha.git && git fetch upstream`.

- Navigate to your `master` branch using `git checkout master`.

- Pull changes from `upstream` using `git pull upstream master`.

- If any changes were pulled in, rebase your branch onto `master` by switching back to your branch (`git checkout <your-branch>`) then rebasing using `git rebase master`.

1. Push your changes to your fork; `git push origin`.

1. In your browser, navigate to [mochajs/mocha](https://github.com/mochajs/mocha). You should see a notification about your recent changes in your fork's branch, with a (green?) button to create a pull request. Click it.

1. Describe your changes in detail here, following the template. Once you're satisfied, submit the form.

1. If you have not signed our [Contributor License Agreement](https://js.foundation/cla), a friendly robot will prompt you to do so. A [CLA](https://cla.js.foundation/mochajs/mocha) (electronic) signature is \*\*required\*\* for all contributions of code to Mocha.

1. Continuous integration checks will run against your changes. The result of these checks will be displayed on your PR.

- If the checks fail, you must address those before the PR is accepted.

- GitHub will indicate if there's a conflict. If this happens, you will need to [rebase](https://help.github.com/articles/about-git-rebase/) your branch onto the `master` branch of the source repository. \*\*Do not `git merge`\*\*.

- (Optional) [Squash](https://help.github.com/articles/about-pull-request-merges/#squash-and-merge-your-pull-request-commits) your changesets. If you have multiple changesets in your PR, they will be squashed upon PR acceptance by the Mocha team.

1. Be patient while your PR is reviewed. This can take a while. We may request changes, but don't be afraid to question them.

1. Your PR might become conflicted with the code in `master`. If this is the case, you will need to [update your PR](#up-to-date) and resolve your conflicts.

1. You don't need to make a new PR to any needed changes. Instead, commit on top of your changes, and push these to your fork's branch. The PR will be updated, and CI will re-run.

Join us in the [contributors' chat](https://gitter.im/mochajs/contributors)!

## :angel: I Just Want To Help

\_Excellent.\_ Here's how:

- \*\*Handy with JavaScript?\*\* Please check out the issues labeled [`help wanted`](https://github.com/mochajs/mocha/issues?q=is%3Aopen+is%3Aissue+label%3A%22help+wanted%22) or [`good-first-issue`](https://github.com/mochajs/mocha/issues?q=is%3Aissue+is%3Aopen+sort%3Aupdated-desc+label%3Agood-first-issue). Try `npx good-first-issue mocha`!

- \*\*Can you write ~~good~~ well?\*\* The [documentation](https://mochajs.org) almost always needs some love. See the [doc-related issues](https://github.com/mochajs/mocha/issues?q=is%3Aopen+is%3Aissue+label%3Adocumentation).

- \*\*Design your thing?\*\* [Our site](https://mochajs.org) needs your magic touch.

- \*\*Familiar with Mocha's codebase?\*\* We could use your help triaging issues and/or reviewing pull requests. Please contact an [org member](https://github.com/orgs/mochajs/people), and we'll chat.

- \*\*Want to build our community?\*\* Mocha has a \_lot\_ of users. We could use your help bringing everyone together in peace and harmony. Please contact an [org member](https://github.com/mochajs/people).

- \*\*You can sell dirt to worms?\*\* Let's raise Mocha's profile in the JavaScript and OSS communities. Please contact an [org member](https://github.com/orgs/mochajs/people)!

- \*\*Wait--you write unit tests for \_fun\_?\*\* A PR which increases coverage is unlikely to be turned down.

- \*\*Are you experienced?\*\* :guitar: If you're a seasoned Mocha user, why not help answer some questions in the [main chat room](https://gitter.im/mochajs/mocha)?