# Contributing to Lerna

> Please note that this project is released with a [Contributor Code of Conduct](./CODE\_OF\_CONDUCT.md).

> By participating in this project you agree to abide by its terms.

First, ensure you have the [latest `npm`](https://docs.npmjs.com/).

To get started with the repo:

```sh

$ git clone git@github.com:lerna/lerna.git && cd lerna

$ npm ci

```

## Code Structure

Currently, the [source](https://github.com/lerna/lerna/tree/master) is split up into a few categories:

\* [utils](https://github.com/lerna/lerna/tree/master/utils): shared packages to run git, npm, fs, and more.

\* [core](https://github.com/lerna/lerna/tree/master/core): basic building blocks, including Package-related abstractions and the command superclass.

\* [commands](https://github.com/lerna/lerna/tree/master/commands): each command has an `initialize` and `execute` function.

\* These commands are consumed as yargs subcommands in [core/cli/index.js](https://github.com/lerna/lerna/blob/master/core/cli/index.js), which is required from the executable [`core/lerna/cli.js`](https://github.com/lerna/lerna/blob/master/core/lerna/cli.js).

## Commands

### Run Unit Tests

```sh

$ npm test

# watch for changes

$ npm test -- --watch

# For a specific file (e.g., in core/command/\_\_tests\_\_/command.test.js)

$ npm test -- --watch core/command

```

By default, `npm test` also runs the linter.

You can skip this by calling `jest` directly:

```sh

$ npx jest

$ npx jest --watch

$ npx jest --config jest.integration.js

# etc

```

### Run Integration Tests

```sh

$ npm run integration

# test a specific file

$ npm run integration -- lerna-publish

# watch for changes

$ npm run integration -- --watch

# watch a specific file

$ npm run integration -- --watch lerna-publish

```

### Linting

```sh

$ npm run lint

```

It's also a good idea to hook up your editor to an eslint plugin.

To fix lint errors from the command line:

```sh

$ npm run lint -- --fix

```

### Local CLI Testing

If you want to test out Lerna on local repos:

```sh

$ npm link

```

This will set your global `lerna` command to the local version.

Note: If the local repo that you are testing in \_already\_ depends on lerna,

you'll need to link your local clone of lerna \_into\_ the target repo:

```sh

# in the target repo

$ npm link lerna

```

### Coverage

If you would like to check test coverage, run the coverage script, then open

`coverage/lcov-report/index.html` in your favorite browser.

```sh

$ npm test -- --coverage

# OS X

$ open coverage/lcov-report/index.html

# Linux

$ xdg-open coverage/lcov-report/index.html

```

### Submitting Pull Requests

This project follows [GitHub's standard forking model](https://guides.github.com/activities/forking/). Please fork the project to submit pull requests.