When it comes to open source, there are different ways you can contribute, all

of which are valuable. Here's few guidelines that should help you as you prepare

your contribution.

## Initial steps

Before you start working on a contribution, create an issue describing what you want to build. It's possible someone else is already working on something similar, or perhaps there is a reason that feature isn't implemented. The maintainers will point you in the right direction.

<!-- ## Submitting a Pull Request

- Fork the repo

- Clone your forked repository: `git clone git@github.com:{your\_username}/zod.git`

- Enter the zod directory: `cd zod`

- Create a new branch off the `master` branch: `git checkout -b your-feature-name`

- Implement your contributions (see the Development section for more information)

- Push your branch to the repo: `git push origin your-feature-name`

- Go to https://github.com/vriad/zod/compare and select the branch you just pushed in the "compare:" dropdown

- Submit the PR. The maintainers will follow up ASAP. -->

## Development

The following steps will get you setup to contribute changes to this repo:

1. Fork this repo.

2. Clone your forked repo: `git clone git@github.com:{your\_username}/zod.git`

3. Run `yarn` to install dependencies.

4. Start playing with the code! You can do some simple experimentation in `src/playground.ts` (see `yarn play` below) or start implementing a feature right away.

### Commands

\*\*`yarn build`\*\*

- deletes `lib` and re-compiles `src` to `lib`

\*\*`yarn test`\*\*

- runs all Jest tests and generates coverage badge

\*\*`yarn testone`\*\*

- runs a single test file

- example: `yarn testone src/\_\_tests\_\_/testFileNameHere.ts`

\*\*`yarn play`\*\*

- executes `src/playground.ts`, watches for changes. useful for experimentation

### Tests

Zod uses Jest for testing. After implementing your contribution, write tests for it. Just create a new file under `src/\_\_tests\_\_` or add additional tests to the appropriate existing file.

Before submitting your PR, run `yarn test` to make sure there are no (unintended) breaking changes.

### Documentation

The Zod documentation lives in the README.md. Be sure to document any API changes you implement.

## License

By contributing your code to the zod GitHub repository, you agree to

license your contribution under the MIT license.