# Contributing to enquirer

First and foremost, thank you! We appreciate that you want to contribute to enquirer, your time is valuable, and your contributions mean a lot to us.

## Important!

By contributing to this project, you:

\* Agree that you have authored 100% of the content

\* Agree that you have the necessary rights to the content

\* Agree that you have received the necessary permissions from your employer to make the contributions (if applicable)

\* Agree that the content you contribute may be provided under the Project license(s)

\* Agree that, if you did not author 100% of the content, the appropriate licenses and copyrights have been added along with any other necessary attribution.

## Getting started

\*\*What does "contributing" mean?\*\*

Creating an issue is the simplest form of contributing to a project. But there are many ways to contribute, including the following:

- Updating or correcting documentation

- Feature requests

- Bug reports

If you'd like to learn more about contributing in general, the [Guide to Idiomatic Contributing](https://github.com/jonschlinkert/idiomatic-contributing) has a lot of useful information.

\*\*Showing support for enquirer\*\*

Please keep in mind that open source software is built by people like you, who spend their free time creating things the rest the community can use.

Don't have time to contribute? No worries, here are some other ways to show your support for enquirer:

- star the [project](https://github.com/enquirer/enquirer)

- tweet your support for enquirer

## Issues

Please only create issues for bug reports or feature requests. Issues discussing any other topics may be closed by the project's maintainers without further explanation.

Do not create issues about bumping dependencies unless a bug has been identified and you can demonstrate that it effects this library.

\*\*Help us to help you\*\*

Remember that we’re here to help, but not to make guesses about what you need help with:

- Whatever bug or issue you're experiencing, assume that it will not be as obvious to the maintainers as it is to you.

- Spell it out completely. Keep in mind that maintainers need to think about \_all potential use cases\_ of a library. It's important that you explain how you're using a library so that maintainers can make that connection and solve the issue.

\_It can't be understated how frustrating and draining it can be to maintainers to have to ask clarifying questions on the most basic things, before it's even possible to start debugging. Please try to make the best use of everyone's time involved, including yourself, by providing this information up front.\_

### Before creating an issue

Please try to determine if the issue is caused by an underlying library, and if so, create the issue there. Sometimes this is difficult to know. We only ask that you attempt to give a reasonable attempt to find out. Oftentimes the readme will have advice about where to go to create issues.

Try to follow these guidelines:

- \*\*Avoid creating issues for implementation help\*\* - It's much better for discoverability, SEO, and semantics - to keep the issue tracker focused on bugs and feature requests - to ask implementation-related questions on [stackoverflow.com][so]

- \*\*Investigate the issue\*\* - Search for exising issues (open or closed) that address the issue, and might have even resolved it already.

- \*\*Check the readme\*\* - oftentimes you will find notes about creating issues, and where to go depending on the type of issue.

- Create the issue in the appropriate repository.

### Creating an issue

Please be as descriptive as possible when creating an issue. Give us the information we need to successfully answer your question or address your issue by answering the following in your issue:

- \*\*description\*\*: (required) What is the bug you're experiencing? How are you using this library/app?

- \*\*OS\*\*: (required) what operating system are you on?

- \*\*version\*\*: (required) please note the version of enquirer are you using

- \*\*error messages\*\*: (required) please paste any error messages into the issue, or a [gist](https://gist.github.com/)

- \*\*extensions, plugins, helpers, etc\*\* (if applicable): please list any extensions you're using

### Closing issues

The original poster or the maintainers of enquirer may close an issue at any time. Typically, but not exclusively, issues are closed when:

- The issue is resolved

- The project's maintainers have determined the issue is out of scope

- An issue is clearly a duplicate of another issue, in which case the duplicate issue will be linked.

- A discussion has clearly run its course

## Next steps

\*\*Tips for creating idiomatic issues\*\*

Spending just a little extra time to review best practices and brush up on your contributing skills will, at minimum, make your issue easier to read, easier to resolve, and more likely to be found by others who have the same or similar issue in the future. At best, it will open up doors and potential career opportunities by helping you be at your best.

The following resources were hand-picked to help you be the most effective contributor you can be:

- The [Guide to Idiomatic Contributing](https://github.com/jonschlinkert/idiomatic-contributing) is a great place for newcomers to start, but there is also information for experienced contributors there.

- Take some time to learn basic markdown. We can't stress this enough. Don't start pasting code into GitHub issues before you've taken a moment to review this [markdown cheatsheet](https://gist.github.com/jonschlinkert/5854601)

- The GitHub guide to [basic markdown](https://help.github.com/articles/markdown-basics/) is another great markdown resource.

- Learn about [GitHub Flavored Markdown](https://help.github.com/articles/github-flavored-markdown/). And if you want to really go above and beyond, read [mastering markdown](https://guides.github.com/features/mastering-markdown/).

At the very least, please try to:

- Use backticks to wrap code. This ensures that it retains its formatting and isn't modified when it's rendered by GitHub, and makes the code more readable to others

- When applicable, use syntax highlighting by adding the correct language name after the first "code fence"

[so]: http://stackoverflow.com/questions/tagged/enquirer