# Contributing to Poetry

First off, thanks for taking the time to contribute!

The following is a set of guidelines for contributing to Poetry on GitHub. These are mostly guidelines, not rules. Use your best judgement, and feel free to propose changes to this document in a pull request.

#### Table of contents

[How to contribute](#how-to-contribute)

\* [Reporting bugs](#reporting-bugs)

\* [Suggesting enhancements](#suggesting-enhancements)

\* [Contributing to code](#contributing-to-code)

## How to contribute

### Reporting bugs

This section guides you through submitting a bug report for Poetry.

Following these guidelines helps maintainers and the community understand your report, reproduce the behavior, and find related reports.

Before creating bug reports, please check [this list](#before-submitting-a-bug-report) to be sure that you need to create one. When you are creating a bug report, please include as many details as possible. Fill out the [required template](https://github.com/python-poetry/poetry/blob/master/.github/ISSUE\_TEMPLATE/---bug-report.md), the information it asks helps the maintainers resolve the issue faster.

> \*\*Note:\*\* If you find a \*\*Closed\*\* issue that seems like it is the same thing that you're experiencing, open a new issue and include a link to the original issue in the body of your new one.

#### Before submitting a bug report

\* \*\*Check the [FAQs on the official website](https://python-poetry.org/docs/faq)\*\* for a list of common questions and problems.

\* \*\*Check that your issue does not already exist in the [issue tracker](https://github.com/python-poetry/poetry/issues)\*\*.

#### How do I submit a bug report?

Bugs are tracked on the [official issue tracker](https://github.com/python-poetry/poetry/issues) where you can create a new one and provide the following information by filling in [the template](https://github.com/python-poetry/poetry/blob/master/.github/ISSUE\_TEMPLATE/---bug-report.md).

Explain the problem and include additional details to help maintainers reproduce the problem:

\* \*\*Use a clear and descriptive title\*\* for the issue to identify the problem.

\* \*\*Describe the exact steps which reproduce the problem\*\* in as many details as possible.

\* \*\*Provide your pyproject.toml file\*\* in a [Gist](https://gist.github.com) after removing potential private information (like private package repositories).

\* \*\*Provide specific examples to demonstrate the steps to reproduce the issue\*\*. Include links to files or GitHub projects, or copy-paste-able snippets, which you use in those examples.

\* \*\*Describe the behavior you observed after following the steps\*\* and point out what exactly is the problem with that behavior.

\* \*\*Explain which behavior you expected to see instead and why.\*\*

\* \*\*If the problem is an unexpected error being raised\*\*, execute the corresponding command in \*\*debug\*\* mode (the `-vvv` option).

Provide more context by answering these questions:

\* \*\*Did the problem start happening recently\*\* (e.g. after updating to a new version of Poetry) or was this always a problem?

\* If the problem started happening recently, \*\*can you reproduce the problem in an older version of Poetry?\*\* What's the most recent version in which the problem doesn't happen?

\* \*\*Can you reliably reproduce the issue?\*\* If not, provide details about how often the problem happens and under which conditions it normally happens.

Include details about your configuration and environment:

\* \*\*Which version of Poetry are you using?\*\* You can get the exact version by running `poetry -V` in your terminal.

\* \*\*Which Python version Poetry has been installed for?\*\* Execute the `debug:info` to get the information.

\* \*\*What's the name and version of the OS you're using\*\*?

### Suggesting enhancements

This section guides you through submitting an enhancement suggestion for Poetry, including completely new features and minor improvements to existing functionality. Following these guidelines helps maintainers and the community understand your suggestion and find related suggestions.

Before creating enhancement suggestions, please check [this list](#before-submitting-an-enhancement-suggestion) as you might find out that you don't need to create one. When you are creating an enhancement suggestion, please [include as many details as possible](#how-do-i-submit-an-enhancement-suggestion). Fill in [the template](https://github.com/python-poetry/poetry/blob/master/.github/ISSUE\_TEMPLATE/---feature-request.md), including the steps that you imagine you would take if the feature you're requesting existed.

#### Before submitting an enhancement suggestion

\* \*\*Check the [FAQs on the official website](https://python-poetry.org/docs/faq)\*\* for a list of common questions and problems.

\* \*\*Check that your issue does not already exist in the [issue tracker](https://github.com/python-poetry/poetry/issues)\*\*.

#### How do I submit an Enhancement suggestion?

Enhancement suggestions are tracked on the [official issue tracker](https://github.com/python-poetry/poetry/issues) where you can create a new one and provide the following information:

\* \*\*Use a clear and descriptive title\*\* for the issue to identify the suggestion.

\* \*\*Provide a step-by-step description of the suggested enhancement\*\* in as many details as possible.

\* \*\*Provide specific examples to demonstrate the steps\*\*..

\* \*\*Describe the current behavior\*\* and \*\*explain which behavior you expected to see instead\*\* and why.

### Contributing to code

#### Local development

You will need Poetry to start contributing on the Poetry codebase. Refer to the [documentation](https://python-poetry.org/docs/#introduction) to start using Poetry.

You will first need to clone the repository using `git` and place yourself in its directory:

```bash

$ git clone git@github.com:python-poetry/poetry.git

$ cd poetry

```

Now, you will need to install the required dependency for Poetry and be sure that the current

tests are passing on your machine:

```bash

$ poetry install

$ poetry run pytest tests/

```

Poetry uses the [black](https://github.com/psf/black) coding style and you must ensure that your

code follows it. If not, the CI will fail and your Pull Request will not be merged.

Similarly, the import statements are sorted with [isort](https://github.com/timothycrosley/isort)

and special care must be taken to respect it. If you don't, the CI will fail as well.

To make sure that you don't accidentally commit code that does not follow the coding style, you can

install a pre-commit hook that will check that everything is in order:

```bash

$ poetry run pre-commit install

```

You can also run it anytime using:

```bash

$ poetry run pre-commit run --all-files

```

Your code must always be accompanied by corresponding tests, if tests are not present your code

will not be merged.

#### Pull requests

\* Fill in [the required template](https://github.com/python-poetry/poetry/blob/master/.github/PULL\_REQUEST\_TEMPLATE.md)

\* Be sure that your pull request contains tests that cover the changed or added code.

\* If your changes warrant a documentation change, the pull request must also update the documentation.