# How to contribute

So you want to contribute to Mycroft?

This should be as easy as possible for you but there are a few things to consider when contributing.

The following guidelines for contribution should be followed if you want to submit a pull request.

## How to Prepare

\* You need a [GitHub account](https://github.com/signup/free)

\* Submit an [issue ticket](https://github.com/MycroftAI/mycroft-core/issues) for your issue if one does not already exist.

\* Describe the issue and include steps to reproduce if it's a bug.

\* Ensure to mention the earliest version that you know is affected.

\* If you are able and want to fix this, fork the repository on GitHub and follow the instructions below.

## Make Changes

1. [Fork the Project](https://help.github.com/articles/fork-a-repo/)

2. Clone onto your local machine and set MycroftAI/mycroft-core as your upstream branch

```

git clone https://github.com/<your-username>/<repo-name>

cd <repo-name>

git remote add upstream https://github.com/MycroftAI/mycroft-core

```

3. If one does not already exist, [create a new issue](https://help.github.com/articles/creating-an-issue/) on the [MycroftAI/mycroft-core Issues Tracker](https://github.com/MycroftAI/mycroft-core/issues)

4. Create a \*\*feature\*\* or \*\*bugfix\*\* branch in your forked repo, based on \*\*dev\*\* with your issue identifier. For example, if your issue identifier is: \*\*issue-123\*\* then you will create either: \*\*feature/issue-123\*\* or \*\*bugfix/issue-123\*\*. Use \*\*feature\*\* prefix for issues related to new functionalities or enhancements and \*\*bugfix\*\* in case of bugs found on the \*\*dev\*\* branch

5. Make sure you stick to the coding style and OO patterns that are used already.

6. Document code using [Google-style docstrings](http://sphinxcontrib-napoleon.readthedocs.io/en/latest/example\_google.html). Our automated documentation tools expect that format. All functions and class methods that are expected to be called externally should include a docstring. (And those that aren't [should be prefixed with a single underscore](https://docs.python.org/2/tutorial/classes.html#private-variables-and-class-local-references)).

7. Make commits in logical units and describe them properly. Use your issue identifier at the very beginning of each commit. For instance:

`git commit -m "Issue-123 - Fixing 'A' sound on Spelling Skill"`

8. Before committing, format your code following the PEP8 rules and organize your imports removing unused libs. To check whether you are following these rules, install pep8 and run `pep8 mycroft test` while in the `mycroft-core` folder. This will check for formatting issues in the `mycroft` and `test` folders.

9. Once you have committed everything and are done with your branch, you have to rebase your code with \*\*dev\*\*. Do the following steps:

1. Make sure you do not have any changes left on your branch

2. Checkout on dev branch and make sure it is up-to-date

3. Checkout your branch and rebase it with dev

4. Resolve any conflicts you have

5. You will have to force your push since the historical base has changed

6. Suggested steps are:

```

git checkout dev

git fetch

git reset --hard origin/dev

git checkout <your\_branch\_name>

git rebase dev

git push -f

```

10. If possible, create unit tests for your changes

\* [Unit Tests for most contributions](https://github.com/MycroftAI/mycroft-core/tree/dev/test)

\* [Intent Tests for new skills](https://mycroft-ai.gitbook.io/docs/#testing-your-skill)

\* We utilize TRAVIS-CI, which will test each pull request. To test locally you can run: `./start-mycroft.sh unittest`

11. Once everything is okay, you can finally [create a Pull Request (PR)](https://help.github.com/articles/using-pull-requests/) on [MycroftAi/mycroft-core](https://github.com/MycroftAI/mycroft-core/pulls) to have your code reviewed and merged.

\*\*Note\*\*: Even if you have write access to the master branch, do not work directly on master!

## Submit Changes

\* Push your changes to a topic branch in your fork of the repository.

\* Open a pull request to the original repository and choose the right original branch you want to patch.

\_Advanced users may install the `hub` gem and use the [`hub pull-request` command](https://github.com/defunkt/hub#git-pull-request).\_

\* If not done in commit messages (which you really should do) please reference and update your issue with the code changes. But \_please do not close the issue yourself\_.

\* Even if you have write access to the repository, do not directly push or merge pull-requests. Let another team member review your pull request and approve.

# Additional Resources

\* [General GitHub documentation](http://help.github.com/)

\* [GitHub pull request documentation](https://help.github.com/articles/about-pull-requests/)

\* [Read the Issue Guidelines by @necolas](https://github.com/necolas/issue-guidelines/blob/master/CONTRIBUTING.md) for more details