# Moviepy's contribution guidelines

## Keeping/Improving code quality

- Respect PEP8 conventions!

- Just the right amount of comments. Try to write auto-documented code (with very explicit variable names).

- If you introduce a new functionality or fix a subtle bug, document it in the docstring/code.

- Moviepy's team adopted [black](https://github.com/psf/black) to autoformat the code. This is enforced for any pull request.

> Tip: use `black -t py36 .` to autoformat your code, or set black as a plugin of your editor.

## Using Github

- Keep the discussions on Github to their minimum. Remember that many people receive these messages.

- Opinions are OK.

- Avoid messages that don't bring the discussion forward ("Thanks", etc.)

- For proper discussions, use the [Moviepy Gitter](https://gitter.im/Movie-py). If these discussions lead to a decision (merge/reject), leave a message on the Moviepy issue that summarizes the reasons of the decision.

- Do not push any commit that changes the API without previous discussion.

## Preparing for development

- Fork moviepy using the button in the top right corner

- Clone your fork: `git clone https://github.com/yourname/moviepy.git`

- Add the main repository as a remote: `git remote add upstream https://github.com/Zulko/moviepy.git`

## Standard workflow

- Pull any changes made to the main repository: `git pull upstream master`

- Push these changes to your fork: `git push`

- Always keep your `master` branch up-to-date with `upstream master` and don't develop features on it

- To make a change

- Create a new local branch: `git checkout -b branchname`

- Make any changes in it

- Make sure that your code still conforms to the formatting standard: `black -t py36 .`

- Run the test suite over it to expose any problems: `python3 setup.py test`

- Push the local branch to your fork on github: `git push -u origin branchname`

- Go to github.com/yourname/moviepy and it will display 'Recently pushed branches' giving you the option to make a Pull Request to the main repo

- Fill in any details for your PR and submit

- The test suite will automatically be ran over your submission

- A moviepy collaborator will review your code, and help you to get it merge-ready

- You don't have to have finished your feature/bugfix before submitting a PR; just mention that it is a work-in-progress