

# Contributing

---

## Issues

If you find an error or have a comment, please start a discussion by submitting an issue on our repositories!

- <https://github.com/montoyamoraga/tiny-trainable-instruments/issues>
- <https://github.com/montoyamoraga/TinyTrainable/issues>

## Pull requests

Here is a step by step guide to make pull requests to this repository.

- Create a free GitHub account
- Fork the repository
- Clone your new repository to your computer

```
git clone https://github.com/your_username/tiny-trainable-instruments.git
```

Optionally, you can also clone the submodules of this repository, with the command

```
git submodule update --init --recursive
```

- Change directory (cd) into the project folder

```
cd tiny-trainable-instruments
```

- Make your changes
- Stage and make a commit to your repository on your computer

```
git add .
```

```
git commit -m "your comment"
```

- Push your commit to your personal fork on GitHub
-

```
git push
```

- Open your repository online
- Open your pull request and wait for comments or approval

## Contributing documentation

For more information about how to contribute documentation to an open source artistic project, we recommend looking at the documentation by the p5.js project, available at

[https://github.com/processing/p5.js/blob/main/contributor\\_docs/contributing\\_documentation.md](https://github.com/processing/p5.js/blob/main/contributor_docs/contributing_documentation.md)

## Adding submodules

If you think there are more repositories we should include as submodules for archival purposes, use the following command, replacing GITPATH with the location of the repository you want to include, and FOLDERPATH with the destination.

```
git submodule add GITPATH FOLDERPATH
```

## Helper scripts

The helper scripts are located on the assets/ folder. To run them, cd to assets/ and then use the following commands

### Compiling code

This script uses arduino-cli for checking the compilation of all examples of TinyTrainable.

```
sh compile-code.sh
```

### Delete metadata

This script uses exiftool to delete the metadata of all pictures in docs/

```
sh delete-metadata.sh
```

### Format code

This script uses clang-format to organize all the code in TinyTrainable

```
sh format-code.sh
```

## Markdown to PDF

This script uses pandoc to convert the documentation from Markdown to PDF format.

```
sh markdown-to-pdf.sh
```