

---

# **PicoMusic Documentation**

*Release 0.1.0.dev4*

**Matthew Scott**

**Mar 19, 2018**



---

# Contents

---

<b>1</b>	<b>Contents</b>	<b>3</b>
1.1	PicoMusic . . . . .	3
1.2	Getting Started . . . . .	3
1.3	Troubleshooting . . . . .	4
1.4	Support . . . . .	4
1.5	Contributing . . . . .	4
1.6	Changelog . . . . .	6
1.7	Authors . . . . .	6
<b>2</b>	<b>Indices and tables</b>	<b>7</b>



PicoMusic is a tool for learning about Python and music, by composing music with Python.



## 1.1 PicoMusic

A tiny audiovisual music tool for Python.

### 1.1.1 Purpose

PicoMusic is a tool for learning about Python and music, by composing music with Python.

### 1.1.2 Design goals

- Simple: to help you learn about programming and music composition.
- Pythonic: transfer what you learn to non-musical projects.
- Expressive: enough to create compelling audiovisual masterpieces.
- Finished projects can be exported and shared.
- Runs comfortably on macOS, Windows, and Linux (including Raspbian on Raspberry Pi 3).

## 1.2 Getting Started

Follow these guides to get started with PicoMusic on your system.

### 1.2.1 MacOS

(to be written)

## 1.2.2 Windows

(to be written)

## 1.2.3 Linux (Ubuntu 18.04)

(to be written)

## 1.2.4 Linux (Raspbian on Raspberry Pi)

(to be written)

## 1.3 Troubleshooting

(to be written)

## 1.4 Support

(to be written)

## 1.5 Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

### 1.5.1 Bug reports

When *reporting a bug* please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

### 1.5.2 Documentation improvements

PicoMusic could always use more documentation, whether as part of the official PicoMusic docs, in docstrings, or even on the web in blog posts, articles, and such.

### 1.5.3 Feature requests and feedback

The best way to send feedback is to file an issue at <https://github.com/metrasynth/picomusic/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.

- Remember that this is a volunteer-driven project, and that code contributions are welcome :)

## 1.5.4 Development

To set up *picomusic* for local development:

1. Fork *picomusic* (look for the “Fork” button).
2. Clone your fork locally:

```
git clone git@github.com:your_name_here/picomusic.git
```

3. Create a branch for local development:

```
git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

4. When you’re done making changes, run all the checks, doc builder and spell checker with *tox* one command:

```
tox
```

5. Commit your changes and push your branch to GitHub:

```
git add .
git commit -m "Your detailed description of your changes."
git push origin name-of-your-bugfix-or-feature
```

6. Submit a pull request through the GitHub website.

## Pull Request Guidelines

If you need some code review or feedback while you’re developing the code just make the pull request.

For merging, you should:

1. Include passing tests (run *tox*)<sup>1</sup>.
2. Update documentation when there’s new API, functionality etc.
3. Add a note to *CHANGELOG.rst* about the changes.
4. Add yourself to *AUTHORS.rst*.

## Tips

To run a subset of tests:

```
tox -e envname -- py.test -k test_myfeature
```

To run all the test environments in *parallel* (you need to *pip install detox*):

```
detox
```

<sup>1</sup> If you don’t have all the necessary python versions available locally you can rely on Travis - it will run the tests for each change you add in the pull request.

It will be slower though ...

## 1.6 Changelog

### 1.6.1 0.1.0 series

This is the initial release of PicoMusic.

### 1.6.2 0.1.0.dev4 (2018-03-18)

#### Fixes

- Fix `Pitch.__repr__()`.

### 1.6.3 0.1.0.dev3 (2018-03-18)

#### Changes

- Simplify `Pitch.__repr__()` for better use within tutorials.
- Add full `Fraction` name to interactive namespace, so that the repr of a `Fraction` instance can be entered to get that fraction.
- Fix `Note.__repr__()` to be consistently formatted.
- Documentation improvements.

#### Fixes

- Reduce size of `BasicKit` by trimming silence from end of samples.

### 1.6.4 0.1.0.dev2 (2018-03-11)

#### Fixes

- Correct the singleton implementation of `StageManager`.
- Add all package requirements.
- Include package requirements in `setup.py`.

### 1.6.5 0.1.0.dev1 (2018-03-11)

- Initial release.

## 1.7 Authors

- Matthew Scott

## CHAPTER 2

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`