pytest-cython

Release 0.1.0

April 17, 2016
Overview

This Pytest plugin allows for the doctesting of C extension modules for Python, specifically created through Cython.
You can install “pytest-cython” via pip from PyPI:

```
pip install pytest-cython
```
Usage

Basic usage:

```bash
py.test --doctest-cython
```

### 3.1 Note

- It is assumed that the C extension modules have been build inplace before running `py.test` and there is a matching Cython `.pyx` file
- The `embedsignature` Cython compiler directive must be set to `True`
Contributions are very welcome. Tests can be run with tox:

```
tox
```
License

• Free software: MIT license

Distributed under the terms of the MIT license, “pytest-cython” is free and open source software
Issues

If you encounter any problems, please file an issue along with a detailed description.
Acknowledgements

This Pytest plugin was generated with Cookiecutter along with @hackbrot’s Cookiecutter-pytest-plugin and @ionelmc’s cookiecutter-pylibrary templates.
Installation

At the command line:

```
pip install pytest-cython
```
To use pytest-cython in a project:

```python
import pytest_cython
```
10.1 pytest_cython
Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

11.1 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.

11.2 Documentation improvements

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

11.3 Feature requests and feedback

The best way to send feedback is to file an issue at https://github.com/lgpage/pytest-cython/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 :)

11.4 Development

To set up pytest-cython for local development:

1. Fork pytest-cython (look for the “Fork” button).
2. Clone your fork locally:
git clone git@github.com:your_name_here/pytest-cython.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.

### 11.4.1 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`).
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`.

### 11.4.2 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
```

---

1 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 ...
Authors

• Logan Page - https://github.com/lgpage
13.1 0.1.0 (2016-04-17)

- First release on PyPI.
Indices and tables

- genindex
- modindex
- search
p

pytest_cython, 19
Index

P
pytest_cython (module), 19