Email To Documentation

Release 0.1.0

Alex Kerney

Oct 01, 2017

Contents

Contents:

CHAPTER 1

Email To

Simplyify sending HTML emails

- Free software: MIT license
- Documentation: https://email-to.readthedocs.io.

Judgement rendered by:

Features

The built in Python modules for sending email are powerful, but require a lot of boilerplate to write an HTML formatted email.

```
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
import smtplib

message = MIMEMultipart('alternative')
message['Subject'] = 'Test'
message['From'] = 'user@gmail.com'
message['To'] = 'someone@else.com'

message.attach(MIMEText('# A Heading\nSomething else in the body', 'plain')
message.attach(MIMEText('<h1 style="color: blue">A Heading</a><p>Something else in␣
↪the body</p>', 'html')

server = smtplib.SMTP('smtp.gmail.com', 587)
server.starttls()
server.login('user@gmail.com', 'password')
server.sendmail('user@gmail.com', 'someone@else.com', message.as_string())
server.quit()
```

With `email_to` sending a simple email becomes much more succint.

```
import email_to

server = email_to.EmailServer('smtp.gmail.com', 587, 'user@gmail.com', 'password')
server.quick_email('someone@else.com', 'Test',
                   ['# A Heading', 'Something else in the body'],
                   style='h1 {color: blue}')
```

`email_to` also supports building a message up, line by line. This is especially useful for monitoring scripts where there may be several different conditions of interest.

```
import email_to

server = email_to.EmailServer('smtp.gmail.com', 587, 'user@gmail.com', 'password')

message = server.message()
message.add('# Oh boy, something went wrong!')
message.add('- The server had a hiccup')
message.add('- The power went out')
message.add('- Blame it on a rogue backhoe')
message.style = 'h1 { color: red}'

message.send('someone@else.com', 'Things did not occur as expected')
```

Additionally if the server details are not known at the beginning of the message, that can be handled easily too.

```
import email_to

message = email_to.Message('# Every thing is ok')
message.add('Everything has been running fine for days.')
message.add('Probably time to build something new and break everything')
message.style = 'h1 { color: green }'

server = email_to.EmailServer('smtp.gmail.com', 587, 'user@gmail.com', 'password')
server.send_message(message, 'someone@else.com', 'Things are awesome')
```

Credits

This package was created with Cookiecutter and the audreyr/cookiecutter-pypackage project template.

CHAPTER 2

Installation

Stable release

To install Email To, run this command in your terminal:

```
$ pip install email_to
```

This is the preferred method to install Email To, as it will always install the most recent stable release.

If you don't have pip installed, this Python installation guide can guide you through the process.

From sources

The sources for Email To can be downloaded from the Github repo.

You can either clone the public repository:

```
$ git clone git://github.com/abkfenris/email_to
```

Or download the tarball:

```
$ curl  -OL https://github.com/abkfenris/email_to/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```

CHAPTER 3

Usage

To use Email To in a project:

```
import email_to
```

CHAPTER 4

Contributing

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

You can contribute in many ways:

Types of Contributions

Report Bugs

Report bugs at https://github.com/abkfenris/email_to/issues.

If you are 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.

Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with "bug" and "help wanted" is open to whoever wants to implement it.

Implement Features

Look through the GitHub issues for features. Anything tagged with "enhancement" and "help wanted" is open to whoever wants to implement it.

Write Documentation

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

Submit Feedback

The best way to send feedback is to file an issue at https://github.com/abkfenris/email_to/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 contributions are welcome :)

Get Started!

Ready to contribute? Here's how to set up *email_to* for local development.

1. Fork the *email_to* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/email_to.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv email_to
$ cd email_to/
$ python setup.py develop
```

4. Create a branch for local development:

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

   Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 email_to tests
$ python setup.py test or py.test
$ tox
```

   To get flake8 and tox, just pip install them into your virtualenv.

6. 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
```

7. Submit a pull request through the GitHub website.

Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.6, 2.7, 3.3, 3.4 and 3.5, and for PyPy. Check https://travis-ci.org/abkfenris/email_to/pull_requests and make sure that the tests pass for all supported Python versions.

Tips

To run a subset of tests:

```
$ py.test tests.test_email_to
```

CHAPTER 5

Credits

Development Lead

- Alex Kerney <abk@mac.com>

Contributors

None yet. Why not be the first?

CHAPTER 6

History

0.1.0 (2017-09-27)

- First release on PyPI.

CHAPTER 7

Indices and tables

- genindex
- modindex
- search