
kcrw.apple*newsDocumentation*

Release 0.2.6

Alec Mitchell

Apr 17, 2020

Contents:

1	Features	3
1.1	Installation	3
1.2	Usage	4
1.3	kcrw.apple_news classes	4
1.4	Contributing	6
1.5	Credits	8
1.6	History	8
2	Indices and tables	11
	Python Module Index	13
	Index	15

`kcrw.apple_news` is a simple library for using the [Apple News API](#) from Python 2 or 3 (≥ 2.7 or ≥ 3.5).

- Free software: MIT license
- Documentation: <https://kcrwapple-news.readthedocs.io>.

Features

- Provides support for making signed API requests to the Apple News Publisher API
- Includes support for “Read Channel”, “Create Article”, “Update Article”, “Read Article Information”, “Delete Article” API calls as well making generic signed requests.
- Provides a command line tool `apple_news_api` for making API calls directly.

1.1 Installation

1.1.1 Stable release

To install `kcrw.apple_news`, run this command in your terminal:

```
$ pip install kcrw.apple_news
```

This is the preferred method to install `kcrw.apple_news`, 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.

1.1.2 From sources

The sources for `kcrw.apple_news` can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/KCRW-org/kcrw.apple_news
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/KCRW-org/kcrw.apple_news/tarball/master
```

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

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

1.2 Usage

To use kcrw.apple_news in a project:

```
from kcrw.apple_news import API

api = API('MY_KEY', 'MY_SECRET', 'MY_CHANNEL_ID')
channel_data = api.channel()
new_article = api.create_article(article_data, metadata, {'asset1.jpg': ...})
updated_article = api.update_article(
    'ARTICLE_ID', metadata, article_data, {'updated_asset.jpg': ...}
)
article = api.read('ARTICLE_ID')
api.delete('ARTICLE_ID')
```

You can also use the included command line tool:

```
$ apple_news_api channel $ apple_news_api create path/to/folder $ apple_news_api updated ARTI-
CLE_ID path/to_folder $ apple_news_api read ARTICLE_ID $ apple_news_api delete ARTICLE_ID
```

1.3 kcrw.apple_news classes

Library for making Apple News API requests

class kcrw.apple_news.api.**API** (*key_id, key_secret, channel_id*)
Apple News API object

__init__ (*key_id, key_secret, channel_id*)

Parameters

- **key_id** (*str*) – Apple News API Key Id
- **key_secret** (*str*) – Apple News API Secret
- **channel_id** (*str*) – Apple News Channel Id

channel_id = None

create_article (*article, metadata=None, assets=None*)

Creates and uploads a multi-part article. See [Create an Article](#)

article and *metadata* should contain JSON-serializable dictionaries according the API specification for [article data](#) and optional [article creation metadata](#). *assets* should contain a dict mapping of filenames to file data for all supplemental assets needed for the article.

Parameters

- **article** (*dict -- or None*) – Article data in JSON serializable python dict
- **metadata** – Optional metadata in JSON serializable python dict
- **assets** (*dict -- or None*) – Dict with filename -> data mapping of additional as-sets

Returns dict – the JSON data returned by the API.

Raises AppleNewsException

delete_article (*identifier*)

Deletes an existing a article. See [Delete an Article](#).

Requires an *identifier* for an existing article.

Parameters **identifier** (*str*) – An identifier for an existing Apple News article

Returns dict – the JSON data returned by the API.

Raises AppleNewsException

key_id = None

key_secret = None

read_article (*identifier*)

Retrieves information about an existing a article. See [Read Article Information](#).

Requires an *identifier* for an existing article.

Parameters **identifier** (*str*) – An identifier for an existing Apple News article

Returns dict – the JSON data returned by the API.

Raises AppleNewsException

read_channel ()

Read basic information about the current Apple News Channel

Returns dict – the JSON data returned by the API.

Raises AppleNewsException

send_request (*method, route, body=None, content_type=None*)

Sends a signed request to the Apple News Publisher API.

Parameters

- **method** (*str*) – The HTTP method for the request (e.g. GET, POST, ...)
- **route** (*str*) – The API route for the request
- **body** (*str*) – The request body
- **content_type** (*str*) – The request content type

Returns dict – the JSON data returned by the API.

Raises AppleNewsException

update_article (*identifier, metadata, article=None, assets=None*)

Updates an existing a article. See [Update an Article](#).

Requires an *identifier* for an existing article, and *metadata* containing the current article revision identifier. All other arguments are optional.

Parameters

- **identifier** (*str*) – An identifier for an existing Apple News article
- **metadata** (*dict*) – JSON formatted article metadata
- **article** (*dict* -- or *None*) – Article data in JSON serializable python dict
- **assets** (*dict* -- or *None*) – Dict with filename -> data mapping of additional assets

Returns dict – the JSON data returned by the API.

Raises AppleNewsException

```
url_base = 'https://news-api.apple.com'
```

```
exception kcrw.apple_news.api.AppleNewsError(*args, **kw)
```

Exception class for errors related to Apple News API requests

```
__init__(*args, **kw)
```

Initialize self. See help(type(self)) for accurate signature.

```
code = None
```

```
data = None
```

```
kcrw.apple_news.api.ensure_binary(s, encoding='utf8')
```

```
kcrw.apple_news.api.ensure_text(s, encoding='utf8')
```

1.4 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:

1.4.1 Types of Contributions

Report Bugs

Report bugs at https://github.com/KCRW-org/kcrw.apple_news/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

kcrw.apple_news could always use more documentation, whether as part of the official kcrw.apple_news 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/KCRW-org/kcrw.apple_news/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 :)

1.4.2 Get Started!

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

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

```
$ git clone git@github.com:your_name_here/kcrw.apple_news.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 kcrw.apple_news
$ cd kcrw.apple_news/
$ 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 kcrw.apple_news tests
$ python setup.py test
$ tox
```

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

If you have specified `cookiecutter.use_pytest == 'y'` run the tests with:

```
$ python setup.py pytest
```

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.

1.4.3 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.

1.4.4 Tips

To run a subset of tests:

```
$ pytest tests.test_kcrw.apple_news
```

1.4.5 Deploying

A reminder for the maintainers on how to deploy. Make sure all your changes are committed (including an entry in HISTORY.rst). Then run:

```
$ bump2version patch # possible: major / minor / patch
$ git push
$ git push --tags
```

1.5 Credits

1.5.1 Development Lead

- Alec Mitchell <alecpm@gmail.com>

1.5.2 Acknowledgments

Development sponsored in part by KCRW radio in Santa Monica, CA. Help fund local public radio by donating at <https://join.kcrw.com>.

The code in this package is largely inspired by the [Apple News API Tutorial](#).

This package relies heavily on [requests](#) and [click](#) and was generated using the [Cookiecutter namespace-template](#) template.

1.6 History

1.6.1 0.2.6 (2020-04-17)

- Fix namespace package issues

1.6.2 0.2.4 (2020-03-09)

- Fix namespace package issues

1.6.3 0.2.3 (2020-02-27)

- Fixes to support applications that pin six < 1.12.0

1.6.4 0.2.2 (2020-02-26)

- Include response error codes and data in custom exception.

1.6.5 0.2.1 (2020-02-25)

- Fix bumpversion error

1.6.6 0.2.0 (2020-02-25)

- Fix botched package release

1.6.7 0.1.0 (2020-02-24)

- First release on PyPI

CHAPTER 2

Indices and tables

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

k

`kcrw.apple_news.api`, 4

Symbols

`__init__()` (*kcrw.apple_news.api.API method*), 4
`__init__()` (*kcrw.apple_news.api.AppleNewsError method*), 6

A

API (class in kcrw.apple_news.api), 4
AppleNewsError, 6

C

channel_id (kcrw.apple_news.api.API attribute), 4
code (kcrw.apple_news.api.AppleNewsError attribute), 6
`create_article()` (*kcrw.apple_news.api.API method*), 4

D

data (kcrw.apple_news.api.AppleNewsError attribute), 6
`delete_article()` (*kcrw.apple_news.api.API method*), 5

E

`ensure_binary()` (*in module kcrw.apple_news.api*), 6
`ensure_text()` (*in module kcrw.apple_news.api*), 6

K

kcrw.apple_news.api (module), 4
key_id (kcrw.apple_news.api.API attribute), 5
key_secret (kcrw.apple_news.api.API attribute), 5

R

`read_article()` (*kcrw.apple_news.api.API method*), 5
`read_channel()` (*kcrw.apple_news.api.API method*), 5

S

`send_request()` (*kcrw.apple_news.api.API method*), 5

U

`update_article()` (*kcrw.apple_news.api.API method*), 5
url_base (kcrw.apple_news.api.API attribute), 6