
newslynx-sc-facebook

Release 0.0.1

Sep 27, 2017

Contents

1	Installation	1
1.1	Production	1
1.2	Development	1
2	Tests	3
3	Documentation	5
4	Continuous Integration	7
5	Contributing	9
6	What's in this module ?	11
7	Contents	13
7.1	Sous Chefs	13

CHAPTER 1

Installation

Production

To install `newslynx-sc-facebook` for an active installation of `newslynx-core`, run the following command:

```
$ newslynx sc-install https://github.com/newslynx/newslynx-sc-facebook.git
```

To add `newslynx-sc-facebook` all orgnaizations, run:

```
$ newslynx sc-sync
```

Development

If you want to modify / add Sous Chefs to `newslynx-sc-facebook`, do the following:

NOTE Will install a fresh version of `newslynx` via `pip`.

```
$ git clone https://github.com/newslynx/newslynx-sc-facebook.git
$ cd newslynx-sc-facebook
$ pip install --editable .
```

You should now be able to run `newslynx-sc-facebook`'s Sous Chefs in development mode

```
% newslynx sc-run newslynx_sc_facebook/say_my_name.yaml --myname='Brian Abelson'
```


CHAPTER 2

Tests

Requires nose

```
$ make all_tests
```


CHAPTER 3

Documentation

Documentation for `newslynx-sc-facebook` is hosted on [Read The Docs](#).

It's generated via the following steps

- converts this file (`README.md`) into a ReStructured Text file, saving it to `docs/index.rst`
- runs `newslynx sc-docs newslynx_sc_facebook -f rst` to generate documentation for all the Sous Chefs in `newslynx-sc-facebook` and saves the output to `docs/sous-chefs.rst`
- Builds Sphinx Documentation from these files.

CHAPTER 4

Continuous Integration

Builds for `newslinx-sc-facebook` can be found on [Travis](#)

CHAPTER 5

Contributing

See the [contributing guidelines](#).

CHAPTER 6

What's in this module ?

- `README.md`
 - This file
- `VERSION`
 - `newsllynx-sc-facebook`'s source-of-truth version.
- `requirements.txt`
 - `newsllynx-sc-facebook`'s python dependencies.
- `MANIFEST.in`
 - Specifications for which files to include in the PyPI distribution.
 - See the docs on this [here](#).
- `setup.py`
 - Specification's for building `newsllynx-sc-facebook`'s PyPI distribution.
- `.travis.yml`
 - Configurations for Travis Continuous Integration
 - You must activate this project on [travis-ci.org](#) for this to run on subsequent updates.
- `Makefile`
 - Helpers for managing `newsllynx-sc-facebook`.
 - Includes:
 - * `make clean`:
 - Cleans out cruft from this directory.
 - * `make install`:
 - Installs `newsllynx-sc-facebook`. Assumes that you're in a virtual environment.
 - * `make all_tests`:

- Runs the tests.
- * make readme
 - Converts this file to `.rst`, including a table of contents, and saves it to `docs/index.rst`
- * make sous_chef_docs
 - Programmatically generates `Sous Chef` documentation by running `newslynx sc-docs newslynx_sc_facebook/ --format=rst > docs/sous-chefs.rst`.
- * make all_docs:
 - Builds the sphinx docs for `newslynx-sc-facebook` by running the above two commands.
- * make view_docs
 - Serves documentation at `localhost:8000`
- * make register:
 - Registers `newslynx-sc-facebook` on `PyPI`.
- * make distribute:
 - Publishes a new version of `newslynx-sc-facebook` to `PyPI`.
- `CONTRIBUTING.md`
- `newslynx_sc_facebook`
 - `newslynx-sc-facebook`'s source code and `Sous Chef` configuration files.
- `docs`
 - Sphinx documentation for `newslynx-sc-facebook`
- `tests`
 - nose tests for `newslynx-sc-facebook`

Sous Chefs

newslynx-sc-facebook provides access to the following Sous Chefs

Facebook Page to Event

- Extracts events from a facebook page.
- This Sous Chef runs the python module `newslynx_sc_facebook.events.Page`.
- API Slug: `facebook-page-to-event`

Usage

Standalone

Run this Sous Chef via the api, passing in arbitrary runtime options, and stream it's output.

```
$ newslynx api sous-chefs cook -d=newslynx_sc_facebook/facebook_page_to_event.yaml --  
↪passthrough **options
```

Run this Sous Chef via the api, and if applicable, send it's output to bulkload.

```
$ newslynx api sous-chefs cook -d=newslynx_sc_facebook/facebook_page_to_event.yaml_  
↪**options
```

Do either of the above two, but pass in a recipe file

```
$ newslynx api sous-chefs cook -d=recipe.yaml
```

Recipes

Add this Sous Chef to your authenticated org

```
$ newslynx api sous-chefs create -d=newslynx_sc_facebook/facebook_page_to_event.yaml
```

Create a Recipe with this Sous Chef with command line options.

```
$ newslynx api recipes create sous_chef=facebook-page-to-event **options
```

Alternatively pass in a recipe file.

```
$ newslynx api recipes create sous_chef=facebook-page-to-event --data=recipe.yaml
```

Save the outputted `id` of this recipe, and execute it via the API. **NOTE** This will place the recipe in a task queue.

```
$ newslynx api recipes cook id=<id>
```

Alternatively, run the Recipe, passing in arbitrary runtime options, and stream it's output: **NOTE** Will not execute the SousChef's `load` method.

```
$ newslynx api recipes cook id=<id> --passthrough **options
```

Development

Pass runtime options to `facebook-page-to-event` and stream output. **NOTE** Will not execute the SousChef's `load` method.

```
$ newslynx sc-run newslynx_sc_facebook/facebook_page_to_event.yaml option=value1
```

Alternatively pass in a recipe file

```
$ newslynx sc-run newslynx_sc_facebook/facebook_page_to_event.yaml --recipe=recipe.  
↪yaml
```

Options

In addition to default recipe options, `facebook-page-to-event` also accepts the following

- `page_id`
 - This is what comes after the “facebook.com/” in the url for your facebook page. Do not include the full URL.
 - **Required**
 - Should be rendered with a `text` form.
 - Accepts inputs of type:
 - * `string`
- `must_link`
 - Only create an event if there is a link to an existing content item.
 - **Required**

- Should be rendered with a `checkbox-single` form.
 - Choose from:
 - * `False`
 - Accepts inputs of type:
 - * `boolean`
 - Defaults to `False`
- `search_query`
 - A search query to apply to the text and urls of posts on the page
 - **Required**
 - Should be rendered with a `text` form.
 - Accepts inputs of type:
 - * `searchstring`
 - Defaults to `None`
- `set_event_title`
 - Set's the title of the resulting events. This can be a python format string which has access to all of an event's top-level keys: IE: "Content from {authors} at {created}."
 - Should be rendered with a `text` form.
 - Accepts inputs of type:
 - * `string`
 - Defaults to `None`
- `set_event_description`
 - Set's the description of the output events. This can be a python format string which has access to all of an event's top-level keys: IE: "{title}."
 - Should be rendered with a `paragraph` form.
 - Accepts inputs of type:
 - * `string`
 - Defaults to `None`
- `event_status`
 - Set the status of the resulting events. Choose from `pending` and `approved`. Defaults to `pending`.
 - Should be rendered with a `select` form.
 - Choose from:
 - * `pending`
 - * `approved`
 - Accepts inputs of type:
 - * `string`
 - Defaults to `pending`
- `set_event_tag_ids`

- A list of Tag IDs or slugs to automatically apply to events created by this recipe.
- Should be rendered with a `checkbox` form.
- Choose from:
- Accepts inputs of type:
 - * `string`
 - * `numeric`
- Defaults to `[]`
- `set_event_content_items`
 - A list of Content Item IDs and Titles to automatically apply to events created by this Recipe.
 - Should be rendered with a `search` form.
 - Choose from:
 - Accepts inputs of type:
 - * `json`
 - Defaults to `[]`

Facebook Page Timeseries Metrics

- Computes a timeseries of of metrics for one or more facebook pages.
- This Sous Chef runs the python module `newslynx_sc_facebook.metrics.Page`.
- API Slug: `facebook-page-to-org-timeseries`

Usage

Standalone

Run this Sous Chef via the api, passing in arbitrary runtime options, and stream it's output.

```
$ newslynx api sous-chefs cook -d=newslynx_sc_facebook/facebook_page_to_org_  
↳timeseries.yaml --passthrough **options
```

Run this Sous Chef via the api, and if applicable, send it's output to bulkload.

```
$ newslynx api sous-chefs cook -d=newslynx_sc_facebook/facebook_page_to_org_  
↳timeseries.yaml **options
```

Do either of the above two, but pass in a recipe file

```
$ newslynx api sous-chefs cook -d=recipe.yaml
```

Recipes

Add this Sous Chef to your authenticated org

```
$ newslynx api sous-chefs create -d=newslynx_sc_facebook/facebook_page_to_org_
↳timeseries.yaml
```

Create a Recipe with this Sous Chef with command line options.

```
$ newslynx api recipes create sous_chef=facebook-page-to-org-timeseries **options
```

Alternatively pass in a recipe file.

```
$ newslynx api recipes create sous_chef=facebook-page-to-org-timeseries --data=recipe.
↳yaml
```

Save the outputted id of this recipe, and execute it via the API. **NOTE** This will place the recipe in a task queue.

```
$ newslynx api recipes cook id=<id>
```

Alternatively, run the Recipe, passing in arbitrary runtime options, and stream it's output: **NOTE** Will not execute the SousChef's load method.

```
$ newslynx api recipes cook id=<id> --passthrough **options
```

Development

Pass runtime options to facebook-page-to-org-timeseries and stream output. **NOTE** Will not execute the SousChef's load method.

```
$ newslynx sc-run newslynx_sc_facebook/facebook_page_to_org_timeseries.yaml_
↳option=value1
```

Alternatively pass in a recipe file

```
$ newslynx sc-run newslynx_sc_facebook/facebook_page_to_org_timeseries.yaml --
↳recipe=recipe.yaml
```

Options

In addition to default recipe options, facebook-page-to-org-timeseries also accepts the following

- page_id
 - This is what comes after the “facebook.com/” in the url for your facebook page. Do not include the full URL.
 - **Required**
 - Should be rendered with a text form.
 - Accepts inputs of type:
 - * string

Metrics

facebook-page-to-org-timeseries generates the following Metrics

- fb_page_likes

- Display name: Facebook Page Likes
- Type: cumulative
- Org Levels:
 - * timeseries
 - * summary