# libslack Documentation

***Release 0.2.0***

**Nicola Coretti**

February 21, 2016

Contents


**1 libslack** **1**
- 1.1 Status . . . 1
- 1.2 Tickets . . . 1
- 1.3 Overview . . . 1
- 1.4 Links . . . 1
- 1.5 Requirements . . . 1
- 1.6 Dependencies . . . 1
- 1.7 Installation . . . 1
  - 1.7.1 From PyPi . . . 1
  - 1.7.2 From Github . . . 2
  - 1.7.3 From Source . . . 2

**2 Tools** **3**
- 2.1 Slack CMD (scmd) . . . 3
  - 2.1.1 Usage . . . 3
  - 2.1.2 Example . . . 3
- 2.2 Slack Shell (sls) . . . 3
  - 2.2.1 Usage . . . 4
  - 2.2.2 Screenshot . . . 4

**3 API-Reference** **5**

**4 License** **7**

**5 Indices and tables** **9**

**Python Module Index** **11**

CHAPTER 1

# libslack

## 1.1 Status

## 1.2 Tickets

## 1.3 Overview

libslack is a lightweight python wrapper around the Slack-Web-Api. As part of libslack a small tool called scmd.py which provides a command line interface for the slack api.

## 1.4 Links

- Project on Github
- Documentation

## 1.5 Requirements

- Python 3.4 and newer

## 1.6 Dependencies

- docopt

## 1.7 Installation

### 1.7.1 From PyPi

```
pip install libslack
```

### 1.7.2 From Github

```
pip install git+https://github.com/Nicoretti/libslack.git
```

### 1.7.3 From Source

```
python3 setup.py install
```

CHAPTER 2

# Tools

## 2.1 Slack CMD (scmd)

The Slack CMD enables its user to use the slack web api via the command line.

### 2.1.1 Usage

```
Usage:

scmd (API_COMMAND | -h | -v ) [<params>] [--auth-token=<token>]

Options:
  -h                    Show this screen.
  -v                    Show version.
  --auth-token=<token>  The authentication token which will be used to access
                        the slack api.
                        As an alternative you can specify it in the .slackrc
                        or set the $SLACK_API_TOKEN environment variable.
```

### 2.1.2 Example

#### Call auth.test method

```
User@Host ~ $ scmd auth.test --auth-token xoxb-xxxxxxxxxx-yyyyyyyyyyyyyyyyyyyyyyyy
{'user_id': 'UXXXXXXXX', 'url': 'https://teamname.slack.com/', 'team': 'Teamname', 'user': 'username
```

#### Send a chat message using the web api

```
User@Host ~ $ scmd chat.postMessage "{'channel': 'CXXXXXXXX', 'text': 'this is a message send fd fro
{'ok': True, 'message': {'username': 'bot', 'subtype': 'bot_message', 'ts': '1445633421.000121', 'te
```

## 2.2 Slack Shell (sls)

The Slack Shell provides an interactive shell which enables the user to interactively interact with the slack web api (e.g. query all registered users).

### 2.2.1 Usage

```
Usage:

  sls [--auth-token=<token>]
  sls -h | --help
  sls -v | --version

Options:
  -h --help             Show this screen.
  -v --version          Show version.
  --auth-token=<token>  The authentication token which will be used to access
                        the slack api.
                        As an alternative you can specify it in the .slackyrc
                        or set the $SLACK_API_TOKEN environment variable.
```

### 2.2.2 Screenshot

```
Slack Shell (SLS)
To get help type help

((SLS)) >>> help

Documented commands (type help <topic>):
========================================
help  list  quit

((SLS)) >>> list help
((SLS)) >>> help list

        The list command can be used to query various information fon the slack server.

        Usage:
            list channels [-h]
            list users [-h]

        Options:

            -h  prints help for the preceding command.

((SLS)) >>>
```

CHAPTER 3

# API-Reference

**class** `libslack.slackapi.`**`SlackApi`**(*authentication_token*, *request=None*)

A SlackApi object can be used to interact with the slack api.

For futher details see methods.

**`call`**(*api_call*, *parameters=None*)

Executes a call on the remote slack api and returns an appropriate `SlackApiResponse` object.

**Parameters**

- **`api_call`** (*`string`*) – which identifies the web api endpoint which shall be called.
- **`parameters`** (*`dict`*) – which will be supplied to the web api endpoint.

  e.g.: `{'channel': 'C1234567890', 'text': 'this message will be sent'}`

  > **Attention:** The authentication-token allways will be passed to the endpoint implicitly.

**Returns** an ApiResponse object containing the data provided by the slackapi endpoint.

**Raise** Exception if an unknown api method is called.

**class** `libslack.slackapi.`**`SlackApiRequest`**(*api_call*, *authentication_token*)

The SlackApiRequest class handles and encapsulates a slack request.

**`execute`**(*request_parameters=None*)

Executes an ApiRequest, request an api call.

**Parameters**

- **`request_parameters`** (*`dict`*) – which will be supplied to the web api endpoint.

  e.g.: `{'channel': 'C1234567890', 'text': 'this message will be sent'}`

  > **Attention:** The authentication-token allways will be passed to the endpoint implicitly.

- **`authentication_token`** – which grants access to the api.

**Returns** an SlackApiResponse object based on the response of the slackapi endpoint.

**Raise** Exception if an error occurs while executing the api call.

**class** `libslack.slackapi.`**`SlackApiResponse`**(*response_data*)

The SlackApiResponse class handles and encapsulates an response provided by the slackapi after an api method was called at the endpoint.

**`get_error_message`**()

If is_error returns True, this method returns a more detail error message.

**Returns** a string which provides a more detail error message.

**`is_error`**()

Indicates whether or not the response indicates that an error occured while trying to execute the associated api call.

**Returns** True if an error has occurred, otherwise False.

CHAPTER 4

# License

Copyright (c) 2015, Nicola Coretti All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS WARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

CHAPTER 5

# Indices and tables

- genindex
- modindex
- search

# Python Module Index

## l

libslack.slackapi, 5

## C

call() (libslack.slackapi.SlackApi method), 5

## E

execute() (libslack.slackapi.SlackApiRequest method), 5

## G

get_error_message() (libslack.slackapi.SlackApiResponse method), 6

## I

is_error() (libslack.slackapi.SlackApiResponse method), 6

## L

libslack.slackapi (module), 5

## S

SlackApi (class in libslack.slackapi), 5
SlackApiRequest (class in libslack.slackapi), 5
SlackApiResponse (class in libslack.slackapi), 5