
slacklog Documentation

Release 0.9.5

Mikko Värri

Oct 02, 2018

Contents

1	slacklog – Convert Slackware Changelog to various formats	3
1.1	Basic usage	3
1.2	Requirements	4
1.3	Installation	4
1.4	Trying it in Docker	5
1.5	Authors	5
1.6	License	5
2	Release history	7
2.1	Version 0.9.5 (2018-10-02)	7
2.2	Version 0.9.4 (2017-09-03)	7
2.3	Version 0.9.3 (2017-08-23)	7
2.4	Version 0.9.2 (2017-05-25)	7
2.5	Version 0.9.1 (2017-05-24)	8
2.6	Version 0.9.0 (2017-04-05)	8
2.7	Version 0.0.9 (2017-04-04)	8
2.8	Version 0.0.8 (2014-09-28)	8
2.9	Version 0.0.7 (2011-06-16)	8
2.10	Version 0.0.6 (2011-03-18)	8
2.11	Version 0.0.5 (2011-03-17)	8
2.12	Version 0.0.4 (2011-03-17)	9
2.13	Version 0.0.3 (2011-02-21)	9
2.14	Version 0.0.2 (2011-01-29)	9
2.15	Version 0.0.1 (2011-01-28)	9
3	Commands	11
4	API documentation	13
4.1	SlackLog models	13
4.2	SlackLog parsers	14
4.3	SlackLog formatters	17
5	Indices and tables	27
	Python Module Index	29

Contents:

slacklog – Convert Slackware Changelog to various formats

slacklog provides programs and a library to convert a Slackware ChangeLogs into other formats. Currently, RSS, Atom, JSON, and PyBlosxom formats are supported.

Download: <http://pypi.python.org/pypi/slacklog>

Source code: <http://github.com/vmj/slacklog>

Builds status: <https://travis-ci.org/vmj/slacklog>

Docs: <http://slacklog.readthedocs.org>

Contents

- *slacklog – Convert Slackware Changelog to various formats*
 - *Basic usage*
 - *Requirements*
 - *Installation*
 - *Trying it in Docker*
 - *Authors*
 - *License*

1.1 Basic usage

Typical usage of the program looks like this:

```
$ slacklog2rss --changelog slackware-current/ChangeLog.txt \  
--encoding iso8859-1 \  
--out ~/public_html/slackware-current.rss \  
--slackware "Slackware current" \  
--rssLink "http://linuxbox.fi/~vmj/slackware-current.rss" \  
--description "Slackware current activity" \  
--managingEditor "vmj@linuxbox.fi (Mikko Värri)" \  
--webMaster "vmj@linuxbox.fi (Mikko Värri)"
```

The included Python library provides the ability to make custom formats easily:

```
#!/usr/bin/env python  
from __future__ import print_function  
import codecs  
import locale  
from slacklog.parsers import SlackLogParser  
  
def read(file):  
    '''Return file contents as Unicode.'''  
    return codecs.open(file, 'r', 'iso8859-1').read()  
  
def write(str):  
    '''Print out in preferred encoding.'''  
    print(str.encode(locale.getpreferredencoding()))  
  
# Parse the ChangeLog  
log = SlackLogParser().parse(read('ChangeLog.txt'))  
  
# Just an example of walking the log tree and print it out  
for entry in log.entries:  
    write(u'[%s] %s\n' % (entry.timestamp.isoformat(), entry.description))  
    for pkg in entry.pkgs:  
        write(u'%s:%s' % (pkg.pkg, pkg.description))
```

Note that slacklog package deals solely in Unicode; parser expects to be given Unicode input and formatters generate Unicode data.

1.2 Requirements

In addition to Python, [python-dateutil](#) is required.

Python versions 2.7 and 3.4 - 3.6 are tested, together with python-dateutil versions 2.1 - 2.7.

1.3 Installation

Use either `pip install slacklog` or download the source archive and use `python setup.py install`.

The source code is available at [Python Package Index \(PyPI\)](#) or, if you want the unreleased version, from [Github](#) git repository.

1.4 Trying it in Docker

Here's one way to hack on this inside a container:

```
$ docker run --rm -it -v $(pwd):/slacklog -w /slacklog python:3.7-alpine3.8 sh
# apk add --no-cache curl
# pip install python-dateutil==2.7.3
# python setup.py install
# sh examples/fetch-changelogs.sh
# sh examples/update-slacklog-rss.sh
```

Obviously, you need Docker installed and working, and I'm assuming you run those commands in the git clone.

1.5 Authors

Original author and current maintainer is Mikko Värri (vmj@linuxbox.fi).

1.6 License

slacklog is Free Software, licensed under GNU General Public License (GPL), version 3 or later. See LICENSE.txt file for details.

2.1 Version 0.9.5 (2018-10-02)

This release adds the JSON formatter.

As for dependencies, support for Python 3.3 was dropped since it had it's end-of-life a year ago. Also, dateutil 2.7 is tested to be good.

2.2 Version 0.9.4 (2017-09-03)

The parser subclass can now overwrite the generation of entry identifier and/or checksum.

Other than that, tests were updated to use python-dateutil 2.6.1, and documentation was fixed.

2.3 Version 0.9.3 (2017-08-23)

This release is mainly bug fixing. RSS and atom formatters learnt to take the feed build time (optionally) as an argument, which makes them more testable.

2.4 Version 0.9.2 (2017-05-25)

This release is backwards incompatible with the previous releases: SlackLogParser and SlackLogFormatter (and sub-classes) have to be instantiated, and instead of using class methods, instance methods and properties have to be used.

2.5 Version 0.9.1 (2017-05-24)

This release adds checksum, identifier, and parent fields to SlackLogEntry. Also, a couple of bugs with recognizing package names was resolved.

2.6 Versions 0.9.0 (2017-04-05)

After almost six years with only maintenance releases, it's time to move to beta.

This release switches from distutils to setuptools, and contains some refactoring to scripts (pure refactoring, no changes in the CLI).

2.7 Version 0.0.9 (2017-04-04)

This release does not add any new functionality.

The dependencies were updated: Python 2.7, 3.3 - 3.6, and python-dateutil 2.1 - 2.6. Support for Python 2.6 was dropped, not because it doesn't work but because Python core team doesn't support it.

The code was formatted according to PEP-8, and the example script was updated to include Slackware versions 14.0 and 14.2.

2.8 Version 0.0.8 (2014-09-28)

This release does not add any new functionality, but includes support for Python 3.

In addition, Slackware{,64} 14.1 was added to the example script, and Travis CI and ReadTheDocs were integrated (see the links at the top of the README).

2.9 Version 0.0.7 (2011-06-16)

This release adds Atom feed formatter, and fixes compatibility issue with recent ChangeLog.txt format change which caused empty entries to be generated. Also, Slackware{,64} 13.37 was added to the example script.

2.10 Version 0.0.6 (2011-03-18)

This release adds documentation.

2.11 Version 0.0.5 (2011-03-17)

This release adds the example script in source distribution, too.

2.12 Version 0.0.4 (2011-03-17)

This release contains better error handling, better compatibility with more feed readers, better support for timezones other than UTC, and an example script suitable for a cron job to update RSS feeds.

2.13 Version 0.0.3 (2011-02-21)

Added PyBlosxom formatter and fixed a couple of issues.

2.14 Version 0.0.2 (2011-01-29)

Packaging cleanups.

2.15 Version 0.0.1 (2011-01-28)

Initial release.

CHAPTER 3

Commands

This package includes a few command line programs that demonstrate the use of the library. Use *-help* option of those commands for more information.

4.1 SlackLog models

SlackLog models represent the ChangeLog.txt after parsing.

class `slacklog.models.SlackLog`

Bases: `object`

Little more than a list of `slacklog.models.SlackLogEntry` objects.

entries = `None`

The list of `slacklog.models.SlackLogEntry` objects. Empty by default.

startsWithSeparator = `None`

Whether the log started with entry separator.

If this is `True`, it implies that the empty element preceding that separator was dropped.

This defaults to `False`.

endsWithSeparator = `None`

Whether the log ended with entry separator.

If this is `True`, it implies that the empty element following that separator was dropped.

This defaults to `False`.

class `slacklog.models.SlackLogEntry` (*timestamp, description, log, checksum=None, identifier=None, parent=None, timezone=None*)

Bases: `object`

An entry in a `slacklog.models.SlackLog`.

timestamp = `None`

A `datetime.datetime` timestamp in UTC.

description = `None`

A unicode description which may be empty.

log = None

Reference to the `slacklog.models.SlackLog` that contains this entry.

checksum = None

A unicode checksum or None.

This should identify the entry by content. Two different logs may have the same entry, but those entries have different parent.

identifier = None

A unicode identifier or None.

This should identify the entry by content and parent.

parent = None

A unicode parent identifier or None.

timezone = None

The original timezone of the entry as `datetime.tzinfo` or None.

pkgs = None

The list of `slacklog.models.SlackLogPkg` objects. Empty by default.

class `slacklog.models.SlackLogPkg(pkg, description, entry)`

Bases: `object`

An entry in a `SlackLogEntry`.

pkg = None

A unicode package identifier.

description = None

A unicode description.

entry = None

Reference to the `slacklog.models.SlackLogEntry` that contains this package.

4.2 SlackLog parsers

SlackLog parser reads a Slackware ChangeLog.txt and builds an in-memory representation of it using SlackLog models.

class `slacklog.parsers.SlackLogParser`

Bases: `object`

Parser for recent (13.x) Slackware ChangeLogs.

quiet = None

If `True`, warnings about date parsing are not printed.

min_date = None

If set to a `datetime.datetime` object, older log entries are ignored (not parsed).

ENTRY = None

Counter of entries (for debugging).

PKG = None

Counter of packages (for debugging).

parse (*data*)

Return the in-memory representation of the data.

Parameters `data` (*unicode*) – the ChangeLog.txt content.

Returns in-memory representation of data

Return type `slacklog.models.SlackLog`

split_log_to_entries (*data*)

Split the ChangeLog.txt into a list of unparsed entries.

Parameters `data` (*unicode*) – the ChangeLog.txt content.

Returns list of unparsed entries, separators removed.

Return type [*unicode*]

parse_entry (*data*, *log*)

Parse a single ChangeLog entry.

Parameters

- `data` (*unicode*) – ChangeLog entry content.
- `log` – in-memory representation that is being parsed.

Type `slacklog.models.SlackLog`

Returns in-memory representation of the ChangeLog entry.

Return type `slacklog.models.SlackLogEntry`

gen_entry_checksum (*data*)

Generate ChangeLog entry checksum from data.

Parameters `data` (*unicode*) – ChangeLog entry content.

Returns Entry checksum.

Return type *unicode*

gen_entry_identifier (*data*, *checksum*, *parent*)

Generate ChangeLog entry identifier from data, checksum, and/or parent identifier.

Parameters

- `data` (*unicode*) – ChangeLog entry content.
- `checksum` (*unicode*) – ChangeLog entry checksum.
- `parent` – Parent entry identifier or None

Returns Entry identifier.

Return type *unicode*

parse_entry_timestamp (*data*)

Parse ChangeLog entry timestamp from data.

Parameters `data` (*unicode*) – ChangeLog entry content.

Returns a two element list: timestamp and the rest of the entry.

Return type [`datetime.datetime`, *unicode*]

parse_entry_description (*data*)

Parse ChangeLog entry description from data.

Parameters `data` (*unicode*) – ChangeLog entry content (without timestamp).

Returns a two element list: description and the rest of the entry.

Return type `[unicode, unicode]`

split_entry_to_pkgs (*data*)

Split ChangeLog entry content into a list of unparsed packages.

Parameters **data** (*unicode*) – ChangeLog entry content (without timestamp or description).

Returns a list of unparsed packages.

Return type `[unicode]`

parse_pkg (*data*, *entry*)

Parse a single package.

Parameters

- **data** (*unicode*) – Package name and description of the update.
- **entry** – in-memory representation of the ChangeLog entry being parsed.

Type `slacklog.models.SlackLogEntry`

Returns in-memory representation of the package.

Return type `slacklog.models.SlackLogPkg`

parse_pkg_name (*data*)

Parse package name from a package.

Parameters **data** (*unicode*) – Package name and description.

Returns a two element list: package name and package description.

Return type `[unicode, unicode]`

parse_pkg_description (*data*)

Parse package description from a package.

Parameters **data** (*unicode*) – Package description.

Returns Package description.

Return type `unicode`

get_line (*data*)

Consume one line from data.

Parameters **data** (*unicode*) – Data.

Returns a two element list: first line, rest of the data.

Return type `[unicode, unicode]`

parse_date (*data*)

Parse a time string into a timestamp.

Parameters **data** (*unicode*) – Time string.

Returns Timestamp in UTC timezone.

Return type `datetime.datetime`

parse_date_with_timezone (*data*)

Parse a time string into a timestamp.

Parameters **data** (*unicode*) – Time string.

Returns a two element list: Timestamp in UTC timezone, and the original timezone.

Return type `datetime.datetime`

4.3 SlackLog formatters

SlackLog formatter takes an in-memory representation of a Slackware ChangeLog.txt and produces a different representation of it.

`slacklog.formatters.readable(d)`

`slacklog.formatters.anchor(d)`

class `slacklog.formatters.SlackLogFormatter`

Bases: `object`

Base class for SlackLog formatters.

This class is ment for subclassing.

max_entries = `None`

If not `None`, must be an `int` representing how many entries are formatted from the beginning of the log. Rest of the entries are ignored.

max_pkgs = `None`

If not `None`, must be an `int` representing how many packages are formatted from the beginning of each entry. Rest of the packages are ignored.

format (*log*)

Return unicode representation of the in-memory representation of the log.

Default implementation calls `format_log_preamble()`, followed by a call to `format_entry()` for each log entry, and finally calls `format_log_postamble()`.

The return value is the concatenation of the return values of the mentioned functions.

Parameters *log* – in-memory representation of the log.

Type `slacklog.models.SlackLog`

Returns Unicode representation of the log.

Type `unicode`

format_log_preamble (*log*)

Return unicode representation of the log preamble, the part before entries.

Default implementation returns empty string.

Parameters *log* – in-memory representation of the log.

Type `slacklog.models.SlackLog`

Returns Unicode representation of log preamble.

Type `unicode`

format_log_postamble (*log*)

Return unicode representation of the log postamble, the part after all entries.

Default implementation returns empty string.

Parameters *log* – in-memory representation of the log.

Type `slacklog.models.SlackLog`

Returns Unicode representation of log postamble.

Type unicode

format_entry (*entry*, *is_first*, *is_last*)

Return unicode representation of a single log entry.

Default implementation calls `format_entry_separator()` with arguments *is_first* and *is_last*, followed by a call to `format_entry_preamble()`, followed by a call to `format_pkg()` for each package in this log entry, finally followed by a call to `format_entry_postamble()`.

The return value is the concatenation of the return values of the mentioned functions.

Parameters

- **entry** – in-memory representation of the log entry.
- **is_first** (*bool*) – True if this is first entry, False otherwise.
- **is_last** (*bool*) – True if this is last entry, False otherwise.

Type `slacklog.models.SlackLogEntry`

Returns Unicode representation of log entry.

Type unicode

format_entry_separator (*is_first*, *is_last*)

Return unicode representation of the log entry separator.

Default implementation returns an empty string.

Parameters

- **is_first** (*bool*) – True if this is first entry, False otherwise.
- **is_last** (*bool*) – True if this is last entry, False otherwise.

Returns Unicode representation of log entry separator.

Type unicode

format_entry_preamble (*entry*)

Return unicode representation of the log entry preamble, the part before packages.

Default implementation returns an empty string.

Parameters **entry** – in-memory representation of the log entry.

Type `slacklog.models.SlackLogEntry`

Returns Unicode representation of log entry preamble.

Type unicode

format_entry_postamble (*entry*)

Return unicode representation of the log entry postamble, the part after packages.

Default implementation returns an empty string.

Parameters **entry** – in-memory representation of the log entry.

Type `slacklog.models.SlackLogEntry`

Returns Unicode representation of log entry postamble.

Type unicode

format_pkg (*pkg, is_first, is_last*)

Return unicode representation of a single log entry package.

Default implementation calls `format_pkg_separator()`, followed by a call to `format_pkg_preamble()`, and finally calls `format_pkg_postamble()`.

The return value is the concatenation of the return values of the mentioned functions.

Parameters

- **pkg** – in-memory representation of the log entry package
- **is_first** (*bool*) – True if this is first package, False otherwise.
- **is_last** (*bool*) – True if this is last package, False otherwise.

Type `slacklog.models.SlackLogPkg`

Returns Unicode representation of log entry package.

Type unicode

format_pkg_separator (*is_first, is_last*)

Return unicode representation of the log entry package separator.

Default implementation returns an empty string.

Parameters

- **is_first** (*bool*) – True if this is first package, False otherwise.
- **is_last** (*bool*) – True if this is last package, False otherwise.

Returns Unicode representation of log entry package separator.

Type unicode

format_pkg_preamble (*pkg*)

Return unicode representation of the log entry package preamble.

Default implementation returns an empty string.

Parameters **pkg** – in-memory representation of the log entry package

Type `slacklog.models.SlackLogPkg`

Returns Unicode representation of log entry package preamble.

Type unicode

format_pkg_postamble (*pkg*)

Return unicode representation of the log entry package postamble.

Default implementation returns an empty string.

Parameters **pkg** – in-memory representation of the log entry package

Type `slacklog.models.SlackLogPkg`

Returns Unicode representation of log entry package postamble.

Type unicode

format_list (*list_of_items, item_formatter, max_items=None*)

Return unicode representation of a list of objects.

This method is not ment for subclassing.

Parameters

- **list_of_items** – List of items to format.
- **item_formatter** – Function that formats one item.
- **max_items** – Maximum number of items to format. If falsy, all items are formatted.

Type list

Type A callable that takes one item as the first positional argument, two booleans *is_first* and *is_last* as second and third positional arguments, and returns a unicode string.

Type int or falsy.

Returns Formatted data.

Return type unicode

class slacklog.formatters.SlackLogTxtFormatter

Bases: *slacklog.formatters.SlackLogFormatter*

Concrete SlackLog formatter that tries to regenerate the original ChangeLog.txt.

format_log_preamble(log)

Overrides *SlackLogFormatter.format_entry_separator()*.

Parameters log – in-memory representation of the log.

Type *slacklog.models.SlackLog*

Returns Unicode representation of log preamble.

Type unicode

format_log_postamble(log)

Overrides *SlackLogFormatter.format_entry_separator()*.

Parameters log – in-memory representation of the log.

Type *slacklog.models.SlackLog*

Returns Unicode representation of log postamble.

Type unicode

format_entry_separator(is_first, is_last)

Overrides *SlackLogFormatter.format_entry_separator()*.

Parameters

- **is_first** (bool) – True if this is first entry, False otherwise.
- **is_last** (bool) – True if this is last entry, False otherwise.

Returns Unicode representation of log entry separator.

Type unicode

format_entry_preamble(entry)

Overrides *SlackLogFormatter.format_entry_preamble()*.

Parameters entry – in-memory representation of the log entry.

Type *slacklog.models.SlackLogEntry*

Returns Unicode representation of log entry preamble.

Type unicode

format_pkg_preamble (*pkg*)

Overrides *SlackLogFormatter.format_pkg_preamble()*.

Parameters *pkg* – in-memory representation of the log entry package

Type *slacklog.models.SlackLogPkg*

Returns Unicode representation of log entry package preamble.

Type unicode

class *slacklog.formatters.SlackLogRssFormatter*

Bases: *slacklog.formatters.SlackLogFormatter*

Concrete SlackLog formatter that generates an RSS feed.

slackware = None

unicode description of the distro version. E.g. ‘Slackware 13.37’ or ‘Slackware64 current’.

rssLink = None

unicode. Full URL of the RSS feed.

webLink = None

unicode. Full URL of the WWW version of the feed.

description = None

unicode description of the feed.

language = None

unicode language identifier. E.g. ‘en’.

managingEditor = None

unicode. Email, and possibly name, of the feed manager. E.g. ‘[jane@doe.net](#) (Jane Doe)’.

webMaster = None

unicode. Email, and possibly name, of the webmaster. E.g. ‘[john@doe.net](#) (John Doe)’.

lastBuildDate = None

datetime.datetime. Timestamp when this feed was last generated. UTC assumed.

format_log_preamble (*log*)

Overrides *SlackLogFormatter.format_log_preamble()*.

Parameters *log* – in-memory representation of the log.

Type *slacklog.models.SlackLog*

Returns Unicode representation of log preamble.

Type unicode

format_log_postamble (*log*)

Overrides *SlackLogFormatter.format_log_postamble()*.

Parameters *log* – in-memory representation of the log.

Type *slacklog.models.SlackLog*

Returns Unicode representation of log postamble.

Type unicode

format_entry_preamble (*entry*)

Overrides *SlackLogFormatter.format_entry_preamble()*.

Parameters *entry* – in-memory representation of the log entry.

Type `slacklog.models.SlackLogEntry`

Returns Unicode representation of log entry preamble.

Type unicode

format_entry_postamble(*entry*)

Overrides `SlackLogFormatter.format_entry_postamble()`.

Parameters **entry** – in-memory representation of the log entry.

Type `slacklog.models.SlackLogEntry`

Returns Unicode representation of log entry postamble.

Type unicode

format_pkg_preamble(*pkg*)

Overrides `SlackLogFormatter.format_pkg_preamble()`.

Parameters **pkg** – in-memory representation of the log entry package

Type `slacklog.models.SlackLogPkg`

Returns Unicode representation of log entry package preamble.

Type unicode

class `slacklog.formatters.SlackLogAtomFormatter`

Bases: `slacklog.formatters.SlackLogFormatter`

Concrete SlackLog formatter that generates an Atom feed.

slackware = **None**

unicode description of the distro version. E.g. ‘Slackware 13.37’ or ‘Slackware64 current’.

link = **None**

unicode. Full URL of the Atom feed.

webLink = **None**

unicode. Full URL of the HTML version.

name = **None**

unicode. Name of the feed author.

email = **None**

unicode. Email of the feed author.

updated = **None**

`datetime.datetime`. Timestamp when this feed was last generated. UTC assumed.

format_log_preamble(*log*)

Overrides `SlackLogFormatter.format_log_preamble()`.

Parameters **log** – in-memory representation of the log.

Type `slacklog.models.SlackLog`

Returns Unicode representation of log preamble.

Type unicode

format_log_postamble(*log*)

Overrides `SlackLogFormatter.format_log_postamble()`.

Parameters **log** – in-memory representation of the log.

Type `slacklog.models.SlackLog`

Returns Unicode representation of log postamble.

Type unicode

format_entry_preamble (*entry*)

Overrides *SlackLogFormatter.format_entry_preamble()*.

Parameters *entry* – in-memory representation of the log entry.

Type *slacklog.models.SlackLogEntry*

Returns Unicode representation of log entry preamble.

Type unicode

format_entry_postamble (*entry*)

Overrides *SlackLogFormatter.format_entry_postamble()*.

Parameters *entry* – in-memory representation of the log entry.

Type *slacklog.models.SlackLogEntry*

Returns Unicode representation of log entry postamble.

Type unicode

format_pkg_preamble (*pkg*)

Overrides *SlackLogFormatter.format_pkg_preamble()*.

Parameters *pkg* – in-memory representation of the log entry package

Type *slacklog.models.SlackLogPkg*

Returns Unicode representation of log entry package preamble.

Type unicode

class *slacklog.formatters.SlackLogPyblosxomFormatter*

Bases: *slacklog.formatters.SlackLogFormatter*

Concrete SlackLog formatter that generates Pyblosxom blog entries.

quiet = None

If True,

slackware = None

unicode description of the distro version. E.g. ‘Slackware 13.37’ or ‘Slackware64 current’.

datadir = None

Blog entry directory.

extension = None

Blog entry filename extension.

encoding = None

Blog entry file encoding.

tags_separator = None

Separator for tags.

pkg_separator = None

Separator for packages.

pyfilemtime = None

If True, a pyfilemtime compatible filenames are generated.

overwrite = None

If True, already existing blog entries are overwritten.

backup = None

If True, already existing blog entries are copied to backups before overwriting.

entry_preamble = None

unicode. HTML to insert before the entry.

entry_postamble = None

unicode. HTML to insert after the entry.

entry_desc_preamble = None

unicode. HTML to insert before the entry description.

entry_desc_postamble = None

unicode. HTML to insert after the entry description.

entry_pkgs_preamble = None

unicode. HTML to insert before the list of packages.

entry_pkgs_postamble = None

unicode. HTML to insert after the list of packages.

pkg_preamble = None

unicode. HTML to insert before a package.

pkg_postamble = None

unicode. HTML to insert after a package.

pkg_name_preamble = None

unicode. HTML to insert before package name.

pkg_name_postamble = None

unicode. HTML to insert after package name.

pkg_desc_preamble = None

unicode. HTML to insert before package description.

pkg_desc_postamble = None

unicode. HTML to insert after package description.

format_entry (entry, is_first, is_last)

Overrides *SlackLogFormatter.format_entry()*.

Parameters

- **entry** – in-memory representation of the log entry.
- **is_first** (*bool*) – True if this is first entry, False otherwise.
- **is_last** (*bool*) – True if this is last entry, False otherwise.

Type *slacklog.models.SlackLogEntry*

Returns Unicode representation of log entry.

Type unicode

format_entry_preamble (entry)

Overrides *SlackLogFormatter.format_entry_preamble()*.

Parameters **entry** – in-memory representation of the log entry.

Type *slacklog.models.SlackLogEntry*

Returns Unicode representation of log entry preamble.

Type unicode

format_entry_postamble(*entry*)

Overrides *SlackLogFormatter.format_entry_postamble()*.

Parameters *entry* – in-memory representation of the log entry.

Type *slacklog.models.SlackLogEntry*

Returns Unicode representation of log entry postamble.

Type unicode

format_pkg_preamble(*pkg*)

Overrides *SlackLogFormatter.format_pkg_preamble()*.

Parameters *pkg* – in-memory representation of the log entry package

Type *slacklog.models.SlackLogPkg*

Returns Unicode representation of log entry package preamble.

Type unicode

format_entry_basename(*entry*)

Return basename for the log entry.

Parameters *entry* – in-memory representation of the log entry.

Type *slacklog.models.SlackLogEntry*

Returns Unicode representation of log entry name

Type unicode

format_entry_title(*entry*)

Return log entry title.

Parameters *entry* – in-memory representation of the log entry.

Type *slacklog.models.SlackLogEntry*

Returns Unicode representation of log entry title

Type unicode

format_entry_tags(*entry*)

Return log entry tags.

Parameters *entry* – in-memory representation of the log entry.

Type *slacklog.models.SlackLogEntry*

Returns Unicode representation of log entry tags

Type unicode

class *slacklog.formatters.SlackLogJsonFormatter*

Bases: *slacklog.formatters.SlackLogFormatter*

Concrete SlackLog formatter that generates JSON dump.

class *SlackLogEncoder*(*skipkeys=False, ensure_ascii=True, check_circular=True, allow_nan=True, sort_keys=False, indent=None, separators=None, encoding='utf-8', default=None*)

Bases: *json.encoder.JSONEncoder*

JSON encoder that knows how to turn a SlackLog into a dict.

default (*o*)

Implement this method in a subclass such that it returns a serializable object for *o*, or calls the base implementation (to raise a `TypeError`).

For example, to support arbitrary iterators, you could implement default like this:

```
def default(self, o):
    try:
        iterable = iter(o)
    except TypeError:
        pass
    else:
        return list(iterable)
    # Let the base class default method raise the TypeError
    return JSONEncoder.default(self, o)
```

indent = `None`

If not `None`, must be an `int` representing how many spaces to indent the array elements and object keys.

format (*log*)

Return unicode representation of the in-memory representation of the log.

Parameters *log* – in-memory representation of the log.

Type `slacklog.models.SlackLog`

Returns Unicode representation of the log.

Type `unicode`

CHAPTER 5

Indices and tables

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

S

`slacklog.formatters`, [17](#)
`slacklog.models`, [13](#)
`slacklog.parsers`, [14](#)

A

anchor() (in module slacklog.formatters), 17

B

backup (slacklog.formatters.SlackLogPybloxomFormatter attribute), 24

C

checksum (slacklog.models.SlackLogEntry attribute), 14

D

datadir (slacklog.formatters.SlackLogPybloxomFormatter attribute), 23

default() (slacklog.formatters.SlackLogJsonFormatter.SlackLogEncoder method), 26

description (slacklog.formatters.SlackLogRssFormatter attribute), 21

description (slacklog.models.SlackLogEntry attribute), 13

description (slacklog.models.SlackLogPkg attribute), 14

E

email (slacklog.formatters.SlackLogAtomFormatter attribute), 22

encoding (slacklog.formatters.SlackLogPybloxomFormatter attribute), 23

endsWithSeparator (slacklog.models.SlackLog attribute), 13

entries (slacklog.models.SlackLog attribute), 13

entry (slacklog.models.SlackLogPkg attribute), 14

ENTRY (slacklog.parsers.SlackLogParser attribute), 14

entry_desc_postamble (slacklog.formatters.SlackLogPybloxomFormatter attribute), 24

entry_desc_preamble (slacklog.formatters.SlackLogPybloxomFormatter attribute), 24

entry_pkgs_postamble (slacklog.formatters.SlackLogPybloxomFormatter attribute), 24

entry_pkgs_preamble (slacklog.formatters.SlackLogPybloxomFormatter attribute), 24

entry_postamble (slacklog.formatters.SlackLogPybloxomFormatter attribute), 24

entry_preamble (slacklog.formatters.SlackLogPybloxomFormatter attribute), 24

extension (slacklog.formatters.SlackLogPybloxomFormatter attribute), 23

F

format() (slacklog.formatters.SlackLogFormatter method), 17

format() (slacklog.formatters.SlackLogJsonFormatter method), 26

format_entry() (slacklog.formatters.SlackLogFormatter method), 18

format_entry() (slacklog.formatters.SlackLogPybloxomFormatter method), 24

format_entry_basename() (slacklog.formatters.SlackLogPybloxomFormatter method), 25

format_entry_postamble() (slacklog.formatters.SlackLogAtomFormatter method), 23

format_entry_postamble() (slacklog.formatters.SlackLogFormatter method), 18

format_entry_postamble() (slacklog.formatters.SlackLogPybloxomFormatter method), 25

format_entry_postamble() (slacklog.formatters.SlackLogRssFormatter method), 22

format_entry_preamble() (slacklog.formatters.SlackLogAtomFormatter method), 23

format_entry_preamble() (slacklog.formatters.SlackLogFormatter method),

18
format_entry_preamble() (slack-
log.formatters.SlackLogPyblosxomFormatter
method), 24
format_entry_preamble() (slack-
log.formatters.SlackLogRssFormatter method),
21
format_entry_preamble() (slack-
log.formatters.SlackLogTxtFormatter method),
20
format_entry_separator() (slack-
log.formatters.SlackLogFormatter method),
18
format_entry_separator() (slack-
log.formatters.SlackLogTxtFormatter method),
20
format_entry_tags() (slack-
log.formatters.SlackLogPyblosxomFormatter
method), 25
format_entry_title() (slack-
log.formatters.SlackLogPyblosxomFormatter
method), 25
format_list() (slacklog.formatters.SlackLogFormatter
method), 19
format_log_postamble() (slack-
log.formatters.SlackLogAtomFormatter
method), 22
format_log_postamble() (slack-
log.formatters.SlackLogFormatter method),
17
format_log_postamble() (slack-
log.formatters.SlackLogRssFormatter method),
21
format_log_postamble() (slack-
log.formatters.SlackLogTxtFormatter method),
20
format_log_preamble() (slack-
log.formatters.SlackLogAtomFormatter
method), 22
format_log_preamble() (slack-
log.formatters.SlackLogFormatter method),
17
format_log_preamble() (slack-
log.formatters.SlackLogRssFormatter method),
21
format_log_preamble() (slack-
log.formatters.SlackLogTxtFormatter method),
20
format_pkg() (slacklog.formatters.SlackLogFormatter
method), 18
format_pkg_postamble() (slack-
log.formatters.SlackLogFormatter method),
19
format_pkg_preamble() (slack-
log.formatters.SlackLogAtomFormatter
method), 23
format_pkg_preamble() (slack-
log.formatters.SlackLogFormatter method),
19
format_pkg_preamble() (slack-
log.formatters.SlackLogPyblosxomFormatter
method), 25
format_pkg_preamble() (slack-
log.formatters.SlackLogRssFormatter method),
22
format_pkg_preamble() (slack-
log.formatters.SlackLogTxtFormatter method),
20
format_pkg_separator() (slack-
log.formatters.SlackLogFormatter method),
19

G

gen_entry_checksum() (slacklog.parsers.SlackLogParser
method), 15
gen_entry_idenfier() (slacklog.parsers.SlackLogParser
method), 15
get_line() (slacklog.parsers.SlackLogParser method), 16

I

identifier (slacklog.models.SlackLogEntry attribute), 14
indent (slacklog.formatters.SlackLogJsonFormatter at-
tribute), 26

L

language (slacklog.formatters.SlackLogRssFormatter at-
tribute), 21
lastBuildDate (slacklog.formatters.SlackLogRssFormatter
attribute), 21
link (slacklog.formatters.SlackLogAtomFormatter
attribute), 22
log (slacklog.models.SlackLogEntry attribute), 13

M

managingEditor (slack-
log.formatters.SlackLogRssFormatter at-
tribute), 21
max_entries (slacklog.formatters.SlackLogFormatter at-
tribute), 17
max_pkgs (slacklog.formatters.SlackLogFormatter at-
tribute), 17
min_date (slacklog.parsers.SlackLogParser attribute), 14

N

name (slacklog.formatters.SlackLogAtomFormatter at-
tribute), 22

O

overwrite (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 23

P

parent (slacklog.models.SlackLogEntry attribute), 14

parse() (slacklog.parsers.SlackLogParser method), 14

parse_date() (slacklog.parsers.SlackLogParser method), 16

parse_date_with_timezone() (slacklog.parsers.SlackLogParser method), 16

parse_entry() (slacklog.parsers.SlackLogParser method), 15

parse_entry_description() (slacklog.parsers.SlackLogParser method), 15

parse_entry_timestamp() (slacklog.parsers.SlackLogParser method), 15

parse_pkg() (slacklog.parsers.SlackLogParser method), 16

parse_pkg_description() (slacklog.parsers.SlackLogParser method), 16

parse_pkg_name() (slacklog.parsers.SlackLogParser method), 16

pkg (slacklog.models.SlackLogPkg attribute), 14

PKG (slacklog.parsers.SlackLogParser attribute), 14

pkg_desc_postamble (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 24

pkg_desc_preamble (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 24

pkg_name_postamble (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 24

pkg_name_preamble (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 24

pkg_postamble (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 24

pkg_preamble (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 24

pkg_separator (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 23

pkgs (slacklog.models.SlackLogEntry attribute), 14

pyfiletime (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 23

Q

quiet (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 23

quiet (slacklog.parsers.SlackLogParser attribute), 14

R

readable() (in module slacklog.formatters), 17

rssLink (slacklog.formatters.SlackLogRssFormatter attribute), 21

S

SlackLog (class in slacklog.models), 13

slacklog.formatters (module), 17

slacklog.models (module), 13

slacklog.parsers (module), 14

SlackLogAtomFormatter (class in slacklog.formatters), 22

SlackLogEntry (class in slacklog.models), 13

SlackLogFormatter (class in slacklog.formatters), 17

SlackLogJsonFormatter (class in slacklog.formatters), 25

SlackLogJsonFormatter.SlackLogEncoder (class in slacklog.formatters), 25

SlackLogParser (class in slacklog.parsers), 14

SlackLogPkg (class in slacklog.models), 14

SlackLogPyblosxomFormatter (class in slacklog.formatters), 23

SlackLogRssFormatter (class in slacklog.formatters), 21

SlackLogTxtFormatter (class in slacklog.formatters), 20

slackware (slacklog.formatters.SlackLogAtomFormatter attribute), 22

slackware (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 23

slackware (slacklog.formatters.SlackLogRssFormatter attribute), 21

split_entry_to_pkgs() (slacklog.parsers.SlackLogParser method), 16

split_log_to_entries() (slacklog.parsers.SlackLogParser method), 15

startsWithSeparator (slacklog.models.SlackLog attribute), 13

T

tags_separator (slacklog.formatters.SlackLogPyblosxomFormatter attribute), 23

timestamp (slacklog.models.SlackLogEntry attribute), 13

timezone (slacklog.models.SlackLogEntry attribute), 14

U

updated (slacklog.formatters.SlackLogAtomFormatter attribute), 22

W

webLink (slacklog.formatters.SlackLogAtomFormatter attribute), 22

webLink (slacklog.formatters.SlackLogRssFormatter attribute), 21

webMaster (slacklog.formatters.SlackLogRssFormatter attribute), 21