
friendlypins Documentation

Release 1.0.4

Kevin S. Phillips

Jun 23, 2019

Contents:

1	friendlypins	3
2	Overview	9
3	Indices and tables	11
	Python Module Index	13
	Index	15

1.1 friendlypins package

1.1.1 Subpackages

friendlypins.scripts package

Submodules

friendlypins.scripts.fpins module

friendlypins.scripts.webp2jpeg module

Command line tool for converting webp images into jpegs

WARNING This is a prototype, that needs to be more fully tested before it should be considered production ready.

`friendlypins.scripts.webp2jpeg.configure_logging(verbosity)`

Configures the global logger for the application

Parameters `verbosity` (*int*) – numeric value for the verbosity level for the log the larger the number, the more verbose the output

`friendlypins.scripts.webp2jpeg.get_args(args)`

Helper method used to parse command line parameters

Parameters `args` (*str*) – optional command line arguments to be parsed if not provided, args will be parsed from the console

Returns parsed arguments

Return type `argparse.Namespace`

`friendlypins.scripts.webp2jpeg.main(args=None)`

Entry point function

Params `str args` sample command line parameters to use when launching the tool used for debug and testing purposes only When not provided, arguments will be parsed from the command line

Returns error code produced by completing the given operation

Return type `int`

Module contents

Command line interfaces for interacting with project APIs

friendlypins.utils package

Submodules

friendlypins.utils.console_actions module

friendlypins.utils.rest_io module

Module contents

Helper scripts used by various parts of the API

1.1.2 Submodules

friendlypins.api module

friendlypins.board module

Primitives for interacting with Pinterest boards

class `friendlypins.board.Board(data, rest_io)`

Bases: `object`

Abstraction around a Pinterest board

Parameters

- **data** (`dict`) – Raw Pinterest API data describing the board
- **rest_io** (`friendlypins.utils.rest_io.RestIO`) – reference to the Pinterest REST API

delete()

Removes this board and all pins attached to it

description

Gets the descriptive text associated with this board

Return type `str`

name

The name of the board

Return type `str`

num_pins

Gets the total number of pins linked to this board

Return type `int`

pins

Generator for iterating over the pins linked to this board

Return type Generator of `friendlypins.pin.Pin`

unique_id

The unique identifier associated with this board

Return type `int`

url

Web address for the UI associated with the dashboard

Return type `str`

friendlypins.headers module**friendlypins.pin module**

Primitives for operating on Pinterest pins

class `friendlypins.pin.Pin` (*data*, *rest_io*)

Bases: `object`

Abstraction around a Pinterest pin

Parameters

- **data** (*dict*) – Raw Pinterest API data describing a pin
- **rest_io** (`friendlypins.utils.rest_io.RestIO`) – reference to the Pinterest REST API

delete()

Removes this pin from it's respective board

link

Source URL containing the original data for the pin

Return type `str`

media_type

Gets descriptor for the type of data stored in the pin's link

Returns None if the type of data associated with the Pin is unknown

Return type `str`

note

Descriptive text associated with pin

Return type `str`

thumbnail

Gets the thumbnail image associated with this pin

Return type `friendlypins.thumbnail.Thumbnail`

unique_id

The unique identifier associated with this pin

Return type `int`

url

Web address for the UI associated with the pin

Return type `str`

friendlypins.thumbnail module

Primitives for operating on Pinterest pins

class `friendlypins.thumbnail.Thumbnail` (*data*)

Bases: `object`

Abstraction around a Pin's thumbnail

Parameters

- **data** (*dict*) – Raw Pinterest API data describing a thumbnail
- **root_url** (*str*) – URL of the Pinterest REST API
- **token** (*str*) – Authentication token for interacting with the API

height

The height of the thumbnail image, in pixels

Return type `int`

url

Source URL where the thumbnail image can be found

Return type `str`

width

The width of the thumbnail image, in pixels

Return type `int`

friendlypins.user module

Interfaces for interacting with Pinterest users

class `friendlypins.user.User` (*data*, *rest_io*)

Bases: `object`

Abstraction around a Pinterest user and their associated data

Parameters

- **data** (*dict*) – JSON data parsed from the API
- **rest_io** (`friendlypins.utils.rest_io.RestIO`) – reference to the Pinterest REST API

boards

Generator for iterating over the boards owned by this user

Return type Generator of `friendlypins.board.Board`

create_board (*name*, *description=None*)

Creates a new board for the currently authenticated user

Parameters

- **name** (*str*) – name for the new board
- **description** (*str*) – optional descriptive text for the board

Returns reference to the newly created board

Return type *friendlypins.board.Board*

first_name

Gets the first name of the user

Return type *str*

last_name

Gets the last name of the user

Return type *str*

name

Gets the name of the user

alias for first_name + last_name

Return type *str*

num_boards

Gets the total number of boards owned by this user

Return type *int*

num_pins

Gets the total number of pins owned by this user

Return type *int*

unique_id

Gets the internal unique ID associated with the user

Return type *int*

url

Gets the URL of the users profile

Return type *str*

friendlypins.version module

1.1.3 Module contents

Package definition for the project

CHAPTER 2

Overview

User friendly Python wrapper for the [Pinterest developer APIs](#). This project is currently in very early development, with only a bare minimum of functionality, intended for some of my personal projects. I'm sharing it here in case someone finds the implementation helpful for their projects. Also, if anyone would like to contribute to this pet project feel free to fork it and send me pull requests for any changes you may like.

Comments, suggestions and bugs may be reported to the project [maintainer](#)

CHAPTER 3

Indices and tables

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

f

- `friendlypins`, 7
- `friendlypins.board`, 4
- `friendlypins.pin`, 5
- `friendlypins.scripts`, 4
- `friendlypins.scripts.webp2jpeg`, 3
- `friendlypins.thumbnail`, 6
- `friendlypins.user`, 6
- `friendlypins.utils`, 4
- `friendlypins.version`, 7

B

Board (*class in friendlypins.board*), 4
boards (*friendlypins.user.User attribute*), 6

C

configure_logging() (*in module friendlypins.scripts.webp2jpeg*), 3
create_board() (*friendlypins.user.User method*), 6

D

delete() (*friendlypins.board.Board method*), 4
delete() (*friendlypins.pin.Pin method*), 5
description (*friendlypins.board.Board attribute*), 4

F

first_name (*friendlypins.user.User attribute*), 7
friendlypins (*module*), 7
friendlypins.board (*module*), 4
friendlypins.pin (*module*), 5
friendlypins.scripts (*module*), 4
friendlypins.scripts.webp2jpeg (*module*), 3
friendlypins.thumbnail (*module*), 6
friendlypins.user (*module*), 6
friendlypins.utils (*module*), 4
friendlypins.version (*module*), 7

G

get_args() (*in module friendlypins.scripts.webp2jpeg*), 3

H

height (*friendlypins.thumbnail.Thumbnail attribute*), 6

L

last_name (*friendlypins.user.User attribute*), 7
link (*friendlypins.pin.Pin attribute*), 5

M

main() (*in module friendlypins.scripts.webp2jpeg*), 3

media_type (*friendlypins.pin.Pin attribute*), 5

N

name (*friendlypins.board.Board attribute*), 4
name (*friendlypins.user.User attribute*), 7
note (*friendlypins.pin.Pin attribute*), 5
num_boards (*friendlypins.user.User attribute*), 7
num_pins (*friendlypins.board.Board attribute*), 4
num_pins (*friendlypins.user.User attribute*), 7

P

Pin (*class in friendlypins.pin*), 5
pins (*friendlypins.board.Board attribute*), 5

T

Thumbnail (*class in friendlypins.thumbnail*), 6
thumbnail (*friendlypins.pin.Pin attribute*), 5

U

unique_id (*friendlypins.board.Board attribute*), 5
unique_id (*friendlypins.pin.Pin attribute*), 5
unique_id (*friendlypins.user.User attribute*), 7
url (*friendlypins.board.Board attribute*), 5
url (*friendlypins.pin.Pin attribute*), 6
url (*friendlypins.thumbnail.Thumbnail attribute*), 6
url (*friendlypins.user.User attribute*), 7
User (*class in friendlypins.user*), 6

W

width (*friendlypins.thumbnail.Thumbnail attribute*), 6