pbwrap Documentation

Release 1.0.6

Michael Tsoukatos

Feb 06, 2023

Contents:

1 Quickstart 3

Python Module Index 9

Index 11

This wrapper is based on **Pastebin API** read their Documentation here for extra information and usage guide.

CHAPTER 1

Quickstart

Use **pip** to install **pbwrap**

```
pip install pbwrap
```

Import **pbwrap** to you script and create a *Pastebin* object.

```
from pbwrap import Pastebin

pb = Pastebin(your_api_key)
```

You are now ready to request things from the Pastebin API

Note: It's reccomended to authenticate as most endpoints require a user_key API is limited without authenticating.

```
pb.authenticate(username, password)
```

Note: More documentation will be added in later releases!Read the Pastebin Class methods to learn how to use all the API endpoints.

1.1 Installation and Usage

Use **pip** to install **pbwrap**

```
pip install pbwrap
```

Import **pbwrap** to you script and create a *Pastebin* object.

```
from pbwrap import Pastebin

pb = Pastebin(your_api_key)
```

You are now ready to request things from the Pastebin API

Note: It's reccomended to authenticate as most endpoints require a user_key API is limited without authenticating.

```
pb.authenticate(username, password)
```

1.2 Pastebin Class

class pbwrap.**Pastebin**(*api_dev_key=None*, *verify_ssl=True*)

Pastebin class represents your communication with the Pastebin API through its functions you can use every API endpoint avalaible.

Most functions require at least an api_dev_key parameter. Functions for manipulating your pastes through the API require an api_user_key.

Instantiate a Pastebin Object

Parameters

- **api_dev_key** (*string*) – Your API Pastebin key
- **verify_ssl** (*bool*) – If *False*, does not verify ssl certificate. Default: True

authenticate(*username*, *password*)

Authenticate through the API login endpoint Your api_user_key attribute is set automatically

Parameters

- **username** (*string*) – Your username
- **password** (*string*) – Your password

Returns your user_id key

Return type string

create_paste(*api_paste_code*, *api_paste_private=0*, *api_paste_name=None*, *api_paste_expire_date=None*, *api_paste_format=None*)

Create a new paste if succesfull return it's url.

Parameters

- **api_paste_code** (*string*) – your paste text
- **api_paste_private** (*int*) – valid values=0(public),1(unlisted),2(private)
- **api_user_name** – your paste name
- **api_paste_expire_date** (*string*) – check documentation for valid values
- **api_paste_format** (*string*) – check documentation for valid values

Returns new paste url

Return type string

create_paste_from_file(*filepath*, *api_paste_private=0*, *api_paste_name=None*, *api_paste_expire_date=None*, *api_paste_format=None*)
Create a new paste from file if succesfull return it's url.

Parameters

- **filepath** (*string*) – the path of the file
- **api_paste_private** (*int*) – valid values=0(public),1(unlisted),2(private)
- **api_user_name** – your paste name
- **api_paste_expire_date** (*string*) – check documentation for valid values
- **api_paste_format** (*string*) – check documentation for valid values

Returns new paste url

Return type string

delete_user_paste(*api_paste_key*)
Deletes a paste created by the user.

Parameters **api_paste_key** (*string*) – the id key of the paste you want to delete

Returns api response

Return type string

general_params()
Returns parameters that should be included in every request

Returns The options to be passed to requests.*

Return type dictionary

get_archive()
Return archive paste link list.Archive contains 25 most recent pastes.

Returns a list of url strings

Return type list

get_raw_paste(*paste_id*)
Return raw string of given paste_id.

get_raw_paste(pasted_id)

Parameters **paste_id** (*string*) – The ID key of the paste

Returns the text of the paste

Return type string

static get_recent_pastes(*limit=50*, *lang=None*)
Return a list containing dictionaries of paste.

Parameters

- **limit** (*int*) – the limit of the items returned defaults to 50
- **lang** (*string*) – return only pastes from certain language defaults to None

Returns list of Paste objects.

Return type list(*Paste*)

get_trending()
Return a list of paste objects created from the most trending pastes

Returns a list of Paste objects

Return type list

get_user_details()
Return user details in a dictionary. Can only be user after authenticating with get_user_id(username, password).

Returns dictionary containing user details

Return type dictionary

get_user_pastes(*api_results_limit=None*)
Return a list of Pastes created from the user

Parameters **api_results_limit** (*int*) – min=1, max=1000

Returns a list of Pastes created from the user

Return type list

get_user_raw_paste(*api_paste_key*)
Return the raw data of a user paste(even private pastes!) as string.

Parameters **api_paste_key** (*string*) – the id key of the paste you want to fetch

Returns the text of the paste

Return type string

static scrape_paste_metadata(*paste_key*)
Return a dictionary containing the metadata of the paste.

Parameters **paste_key** (*string*) – the unique key of the paste you want to scrape

Returns dictionary containing the metadata of the paste

Return type dictionary

static scrape_raw_paste(*paste_key*)
Return a string containing the text of the paste.

Parameters **paste_key** (*string*) – the unique key of the paste you want to scrape

Returns raw string containing the text of the paste

Return type string

1.3 Formatter Module

Contains function that manipulate the API returns for easier data manipulation.

pbwrap.formatter.**archive_url_format**(*archive_html*)
Return a list with recent pastes ids

Parameters **archive_html** (*string*) – raw html of the archive url

Returns a list containing paste ids

Return type list

pbwrap.formatter.**paste_list_from_xml**(*xml_paste*)
Input an xml list and return a list of paste objects.

Parameters **xml_paste** – An xml formatted response containing Paste information

Type string

Returns a list of Paste objects parsed from the input xml formatted string

Return type list(*Paste*)

pbwrap.formatter.**user_from_xml**(*user_xml_string*)
Return user dictionary from an xml format string

Parameters **user_xml_string** (*string*) – xml formatted string containing user information

Returns A dictionary containing the user info

Return type *User*

1.4 Models Classes

class pbwrap.models.**Paste**(*paste_dict*)
Defines a Paste from Pastebin paste contains the following fields: key, date, title, size, expire_date, private, format_short, format_long, url, hits.

get_raw_text()
Fetch the text of a paste via the public API. :returns: the paste's text :rtype: string, None

scrape_raw_text()
Fetch the ext of a paste via the Paid API. :returns: the paste's text :rtype: string, None

class pbwrap.models.**User**(*user_dict*)
Defines a user contains the following fields: name format_short expiration avatar_url private website email location account_type

Python Module Index

p

pbwrap.formatter, 6

Index

A

archive_url_format() (*in module pbwrap.formatter*), 6
authenticate() (*pbwrap.Pastebin method*), 4

C

create_paste() (*pbwrap.Pastebin method*), 4
create_paste_from_file() (*pbwrap.Pastebin method*), 4

D

delete_user_paste() (*pbwrap.Pastebin method*), 5

G

general_params() (*pbwrap.Pastebin method*), 5
get_archive() (*pbwrap.Pastebin method*), 5
get_raw_paste() (*pbwrap.Pastebin method*), 5
get_raw_text() (*pbwrap.models.Paste method*), 7
get_recent_pastes() (*pbwrap.Pastebin static method*), 5
get_trending() (*pbwrap.Pastebin method*), 5
get_user_details() (*pbwrap.Pastebin method*), 6
get_user_pastes() (*pbwrap.Pastebin method*), 6
get_user_raw_paste() (*pbwrap.Pastebin method*), 6

P

Paste (*class in pbwrap.models*), 7
paste_list_from_xml() (*in module pbwrap.formatter*), 6
Pastebin (*class in pbwrap*), 4
pbwrap.formatter (*module*), 6

S

scrape_paste_metadata() (*pbwrap.Pastebin static method*), 6
scrape_raw_paste() (*pbwrap.Pastebin static method*), 6
scrape_raw_text() (*pbwrap.models.Paste method*), 7

U

User (*class in pbwrap.models*), 7
user_from_xml() (*in module pbwrap.formatter*), 7