
Python WordPress Documentation

Release 1.0.0

Python WordPress Library

May 10, 2017

Contents

1	Features	3
2	Quick Start	5
3	Contents:	7
3.1	Installation	7
3.2	Usage	8
3.3	wordpress	8
3.4	Contributing	28
3.5	Credits	30
3.6	History	30
3.7	Indices and tables	30
	Python Module Index	33

A Python library for interacting with WordPress REST API.

- Free software: MIT License

CHAPTER 1

Features

- TODO

CHAPTER 2

Quick Start

```
>>> from wordpress import WordPress
>>> wp = WordPress('http://example.org/')
>>> posts = wp.list_posts()
>>> for p in posts:
...     print('{title[rendered]}: {link}'.format(**p._json))
Hello, World: http://example.org/2017/01/30/hello-world
```


Installation

Stable release

To install Python WordPress, run this command in your terminal:

```
$ pip install python-wp
```

This is the preferred method to install Python WordPress, as it will always install the most recent stable release.

If you don't have `pip` installed, this [Python installation guide](#) can guide you through the process.

From sources

The sources for Python WordPress can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/myles/python-wp
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/myles/python-wp/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```

Usage

To use Python WordPress in a project:

```
from wordpress import WordPress
wp = WordPress('http://wordpress-site.dev/')
```

Get a list of posts:

```
wp.list_posts()
```

Retrieve a post:

```
wp.get_post(443)
```

Delete a post:

```
wp.delete_post(443)
```

wordpress

wordpress package

Submodules

wordpress.api module

class `wordpress.api.WordPress` (*url*, *verify_ssl=True*)

Bases: `object`

create_comment (***kwargs*)

create_media (***kwargs*)

create_page (***kwargs*)

create_post (*date=None*, *date_gmt=None*, *slug=None*, *status=None*, *password=None*, *title=None*, *content=None*, *author=None*, *excerpt=None*, *featured_media=None*, *comment_status=None*, *ping_status=None*, *format=None*, *meta=None*, *sticky=None*, *template=None*, *categories=None*, *tags=None*, *liveblog_links=None*)

Create a Post.

Parameters

- **date** (*datetime*) – The date the object was published, in the site’s timezone.
- **date_gmt** (*datetime*) – The date the object was published, as GMT.
- **slug** (*str*) – An alphanumeric identifier for the object unique to its type.
- **status** (*str*) – A named status for the object.
One of: `publish`, `future`, `draft`, `pending`, `private`
- **password** (*str*) – A password to protect access to the content and excerpt.
- **title** (*str*) – The title for the object.
- **content** (*str*) – The content for the object.

- **author** (*id*) – The ID for the author of the object.
- **excerpt** (*str*) – The excerpt for the object.
- **featured_media** (*int*) – The ID of the featured media for the object.
- **comment_status** (*str*) – Whether or not comments are open on the object.
One of: open, closed
- **ping_status** (*str*) – Whether or not the object can be pinged.
One of: open, closed
- **format** (*str*) – The format for the object.
One of: standard
- **meta** (*dict*) – Meta fields.
- **sticky** (*bool*) – Whether or not the object should be treated as sticky.
- **template** (*str*) – The theme file to use to display the object.
One of:
- **categories** (*str*) – The terms assigned to the object in the category taxonomy.
- **tags** (*str*) – The terms assigned to the object in the post_tag taxonomy.
- **liveblog_likes** (*str*) – The number of Liveblog Likes the post has.

create_tag (***kwargs*)

create_user (***kwargs*)

delete_comment (***kwargs*)

delete_media (***kwargs*)

delete_page (***kwargs*)

delete_post (*pk, force=False*)

Delete a Post.

Parameters

- **pk** (*int*) – The post id you want to delete.
- **force** (*bool*) – Whether to bypass trash and force deletion.

delete_post_revision (*parent, pk*)

Delete Post Revision.

Parameters

- **parent** (*int/wordpress.models.Post/wordpress.models.Page*) – The id for the parent of the object.
- **pk** (*int*) – Unique identifier for the object.

Returns

Return type dict

delete_tag (***kwargs*)

delete_user (***kwargs*)

get_category (*pk*, *context*='view')

Retrieve a Category.

Parameters

- **pk** (*int*) – The category id you want to retrieve.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

Returns

Return type *wordpress.models.Category*

get_comment (***kwargs*)

get_media (***kwargs*)

get_page (***kwargs*)

get_post (*pk*, *context*='view', *password*=None)

Retrieve a Post.

Parameters

- **pk** (*int*) – The post id you want to retrieve.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

- **password** (*str*) – The password for the post if it is password protected.

Returns

Return type *wordpress.models.Post*

get_post_revision (*parent*, *pk*, *context*='view')

Get a Post Revision.

Parameters

- **parent** (*int/wordpress.models.Post/wordpress.models.Page*) – The id for the parent of the object.
- **pk** (*int*) – Unique identifier for the object.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view

Returns

Return type *wordpress.models.PostRevision*

get_post_status (*slug*, *context*='view')

Retrieve a Post statuses

Parameters

- **slug** (*str*) – The name of the status.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

Returns

Return type *wordpress.models.PostStatus*

get_post_type (***kwargs*)

get_tag (*pk*, *context*='view')

Retrieve a Tag.

Parameters

- **pk** (*int*) – The tag id you want to retrieve.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

Returns

Return type *wordpress.models.Tag*

get_taxonomy (***kwargs*)

get_user (***kwargs*)

list_categories (*context*='view', *page*=1, *pre_page*=10, *search*=None, *exclude*=None, *include*=None, *order*='asc', *orderby*='name', *hide_empty*=False, *parent*=None, *post*=None, *slug*=None)

Get a list of categories.

Parameters

- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

- **page** (*int*) – Current page of the collection.

Default: 1

- **pre_page** (*int*) – Maximum number of items to be returned in result set.

Default: 10

- **search** (*str*) – Limit results to those matching a string.

- **exclude** (*int*) – Ensure result set excludes specific IDs.

Default:

- **include** (*int*) – Limit result set to specific IDs.

Default:

- **order** (*str*) – Order sort attribute ascending or descending.
Default: asc
One of: asc, desc
- **orderby** (*str*) – Sort collection by term attribute.
Default: name
One of: id, include, name, slug, term_group, description, count
- **hide_empty** (*bool*) – Whether to hide terms not assigned to any posts.
- **parent** (*int/wordpress.models.Category*) – Limit result set to terms assigned to a specific parent.
- **post** (*int/wordpress.models.Post*) – Limit result set to terms assigned to a specific post.
- **slug** (*str*) – Limit result set to terms with a specific slug.

Returns A list of `wordpress.models.Category`.

Return type list

`list_comments` (***kwargs*)

`list_media` (***kwargs*)

`list_pages` (***kwargs*)

`list_post_revisions` (*parent, context='view'*)

List Post Revisions.

Parameters

- **parent** (*int/wordpress.models.Post/wordpress.models.Page*) – The id for the parent of the object.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.
Default: view
One of: view

Returns A list of `wordpress.models.PostRevision`.

Return type list

`list_post_statuses` (*context='view'*)

Get a list of post statuses.

Parameters **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

Returns A list of `wordpress.models.PostStatus`

Return type list

`list_post_types` (***kwargs*)


```
list_posts (context='view', page=1, pre_page=10, search=None, after=None, author=None,
            author_exclude=None, before=None, exclude=None, include=None, offset=None,
            order='desc', orderby='date', slug=None, status='publish', categories=None,
            categories_exclude=None, tags=None, tags_exclude=None, sticky=None)
```

Get a list of posts.

Parameters

- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

- **page** (*int*) – Current page of the collection.
- **pre_page** (*int*) – Maximum number of items to be returned in result set.
- **search** (*str*) – Limit results to those matching a string.
- **after** (*datetime*) – Limit response to posts published after a given date.
- **author** (*int*) – Limit result set to posts assigned to specific authors.
- **author_exclude** (*int*) – Ensure result set excludes posts assigned to specific authors.
- **before** (*datetime*) – Limit response to posts published before a given date.
- **exclude** (*int*) – Ensure result set excludes specific IDs.
- **include** (*int*) – Limit result set to specific IDs.
- **offset** (*int*) – Offset the result set by a specific number of items.
- **order** (*str*) – Order sort attribute ascending or descending.

Default: desc

One of: asc, desc

- **orderby** (*str*) – Sort collection by object attribute. Default: date One of: date, relevance, id, include, title, slug
- **slug** (*str*) – Limit result set to posts with one or more specific slugs.
- **status** (*str*) – Limit result set to posts assigned one or more statuses.

Default: publish

One of: publish, future, draft, pending, private

- **categories** (*str*) – Limit result set to all items that have the specified term assigned in the categories taxonomy.
- **categories_exclude** (*str*) – Limit result set to all items except those that have the specified term assigned in the categories taxonomy.
- **tags** (*str*) – Limit result set to all items that have the specified term assigned in the tags taxonomy.
- **tags_exclude** (*str*) – Limit result set to all items except those that have the specified term assigned in the tags taxonomy.
- **sticky** (*bool*) – Limit result set to items that are sticky.

Returns A list of `wordpress.models.Post`.

Return type list

list_tags (*context='view', page=1, pre_page=10, search=None, include=[], offset=0, order='asc', orderby='name', hide_empty=False, post=None, slug=None*)

Get a list of tags.

Parameters

- **context** (*str*) – Scope under which the request is made; determines fields present in response.
Default: view
One of: view, embed, edit
- **page** (*int*) – Current page of the collection.
Default: 1
- **per_page** (*int*) – Maximum number of items to be returned in result set.
Default: 10
- **search** (*str*) – Limit results to those matching a string.
- **exclude** (*str*) – Ensure result set excludes specific IDs.
Default:
- **include** (*list*) – Limit result set to specific IDs.
Default:
- **offset** (*int*) – Offset the result set by a specific number of items.
- **order** (*str*) – Order sort attribute ascending or descending.
Default: asc
One of: asc, desc
- **orderby** (*str*) – Sort collection by term attribute.
Default: name
One of: id, include, name, slug, term_group, description, count
- **hide_empty** (*bool*) – Whether to hide terms not assigned to any posts.
- **post** (*int*) – Limit result set to terms assigned to a specific post.
- **slug** (*str*) – Limit result set to terms with a specific slug.

Returns A list of `wordpress.models.Category`.

Return type list

list_taxonomies (***kwargs*)

list_users (***kwargs*)

update_comment (***kwargs*)

update_media (***kwargs*)

update_page (***kwargs*)

update_post (*pk, date=None, date_gmt=None, slug=None, status=None, password=None, title=None, content=None, author=None, excerpt=None, featured_media=None, comment_status=None, ping_status=None, format=None, meta=None, sticky=None, template=None, categories=None, tags=None, liveblog_links=None*)

Update a Post.

Parameters

- **pk** (*int*) – The ID of the post you want to update.
- **date** (*datetime*) – The date the object was published, in the site’s timezone.
- **date_gmt** (*datetime*) – The date the object was published, as GMT.
- **slug** (*str*) – An alphanumeric identifier for the object unique to its type.
- **status** (*str*) – A named status for the object.
One of: publish, future, draft, pending, private
- **password** (*str*) – A password to protect access to the content and excerpt.
- **title** (*str*) – The title for the object.
- **content** (*str*) – The content for the object.
- **author** (*id*) – The ID for the author of the object.
- **excerpt** (*str*) – The excerpt for the object.
- **featured_media** (*int*) – The ID of the featured media for the object.
- **comment_status** (*str*) – Whether or not comments are open on the object.
One of: open, closed
- **ping_status** (*str*) – Whether or not the object can be pinged.
One of: open, closed
- **format** (*str*) – The format for the object.
One of: standard
- **meta** (*dict*) – Meta fields.
- **sticky** (*bool*) – Whether or not the object should be treated as sticky.
- **template** (*str*) – The theme file to use to display the object.
One of:
- **categories** (*str*) – The terms assigned to the object in the category taxonomy.
- **tags** (*str*) – The terms assigned to the object in the post_tag taxonomy.
- **liveblog_likes** (*str*) – The number of Liveblog Likes the post has.

update_setting (*title=None, description=None, url=None, email=None, timezone=None, date_format=None, time_format=None, start_of_week=None, language=None, use_smilies=None, default_category=None, default_post_format=None, post_pre_page=None*)

Update WordPress settings.

Parameters

- **title** (*str*) – Site title.
- **description** (*str*) – Site description.
- **url** (*str*) – Site URL.
- **email** (*str*) – This address is used for admin purposes. If you change this we will send you an email at your new address to confirm it. The new address will not become active until confirmed.

- **timezone** (*str*) – A city in the same timezone as you.
- **date_format** (*str*) – A date format for all date strings.
- **time_format** (*str*) – A time format for all time strings.
- **start_of_week** (*int*) – A day number of the week that the week should start on.
- **language** (*str*) – WordPress locale code.
- **use_smilies** (*bool*) – Convert emoticons like :-) and :-P to graphics on display.
- **default_category** (*int*) – Default category.
- **default_post_format** (*str*) – Default post format.
- **posts_per_page** (*int*) – Blog pages show at most.

`update_tag` (***kwargs*)

`update_user` (***kwargs*)

wordpress.models module

Copyright 2009-2010 Joshua Roeslein

class `wordpress.models.Category` (*api=None*)

Bases: `wordpress.models.Model`

A WordPress Category

Parameters

- **id** (*int*) – Unique identifier for the term.
Context: view, embed, edit
- **count** (*int*) – Number of published posts for the term.
Context: view, edit
- **description** (*str*) – HTML description of the term.
Context: view, edit
- **link** (*str*) – URL of the term.
Context: view, embed, edit
- **name** (*str*) – HTML title for the term.
Context: view, embed, edit
- **slug** (*str*) – An alphanumeric identifier for the term unique to its type.
Context: view, embed, edit
- **taxonomy** (*str*) – Type attribution for the term.
Context: view, embed, edit
One of: category, post_tag, nav_menu, link_category, post_format
- **parent** (*int*) – The parent term ID.
Context: view, edit

- **meta** (*dict*) – Meta fields.

Context: view, edit

delete (***kwargs*)

classmethod parse (*api, json*)

update (***kwargs*)

class `wordpress.models.Comment` (*api=None*)

Bases: `wordpress.models.Model`

class `wordpress.models.Media` (*api=None*)

Bases: `wordpress.models.Model`

class `wordpress.models.Model` (*api=None*)

Bases: `object`

classmethod parse (*api, json*)

Parse a JSON object into a model instance.

classmethod parse_list (*api, json_list*)

Parse a list of JSON objects into a result set of model instances.

class `wordpress.models.Page` (*api=None*)

Bases: `wordpress.models.Model`

class `wordpress.models.Post` (*api=None*)

Bases: `wordpress.models.Model`

A WordPress post object.

Parameters

- **date** (*datetime*) – The date the object was published, in the site’s timezone.
- **date_gmt** (*datetime*) – The date the object was published, as GMT.
- **guid** (*dict*) – The globally unique identifier for the object.
- **id** (*int*) – Unique identifier for the object.
- **link** (*str*) – URL to the object.
- **modified** (*datetime*) – The date the object was last modified, in the site’s timezone.
- **modified_gmt** (*datetime*) – The date the object was last modified, as GMT.
- **slug** (*str*) – An alphanumeric identifier for the object unique to its type.
- **status** (*str*) – A named status for the object.
- **type** (*str*) – Type of Post for the object.
- **password** (*str*) – A password to protect access to the content and excerpt.
- **title** (*dict*) – The title for the object.
- **content** (*dict*) – The content for the object.
- **author** (*int*) – The ID for the author of the object.
- **excerpt** (*dict*) – The excerpt for the object.
- **featured_media** (*int*) – The ID of the featured media for the object.
- **comment_status** (*str*) – Whether or not comments are open on the object.

- **ping_status** (*str*) – Whether or not the object can be pinged.
- **format** (*str*) – The format for the object.
- **meta** (*list*) – Meta fields.
- **sticky** (*bool*) – Whether or not the object should be treated as sticky.
- **template** (*str*) – The theme file to use to display the object.
- **categories** (*list*) – The terms assigned to the object in the category taxonomy.
- **tags** (*list*) – The terms assigned to the object in the post_tag taxonomy.
- **liveblog_likes** (*int*) – The number of Liveblog Likes the post has.

delete (***kwargs*)
Delete the post.

classmethod parse (*api, json*)
Parse a JSON object into a model instance.

revision (*pk, **kwargs*)
Get a revision of the post.

revisions (***kwargs*)
Lookup revisions of the post.

update (***kwargs*)
Update the post.

class `wordpress.models.PostRevision` (*api=None*)

Bases: `wordpress.models.Model`

A WordPress post object.

Parameters

- **author** (*int*) – The id for the author of the revision.
Context: view
- **date** (*datetime*) – The date the object was published.
Context: view
- **date_gmt** (*datetime*) – The date the object was published, as GMT.
Context: view
- **guid** (*dict*) – GUID for the object, as it exists in the database.
Context: view
- **id** (*int*) – Unique identifier for the object.
Context: view
- **modified** (*datetime*) – The date the object was last modified.
Context: view
- **modified_gmt** (*datetime*) – The date the object was last modified, as GMT.
Context: view
- **parent** (*int*) – The id for the parent of the object.
Context: view

- **slug** (*str*) – An alphanumeric identifier for the object unique to its type.

Context: view

- **title** (*str*) – Title for the object, as it exists in the database.

Context: view

- **content** (*str*) – Content for the object, as it exists in the database.

Context: view

- **excerpt** (*str*) – Excerpt for the object, as it exists in the database.

Context: view

destroy (***kwargs*)

classmethod parse (*api, json*)

class `wordpress.models.PostStatus` (*api=None*)

Bases: `wordpress.models.Model`

A WordPress post object.

Parameters

- **name** (*str*) – The title for the resource.
Context: embed, view, edit
- **private** (*bool*) – Whether posts with this resource should be private.
Context: edit
- **protected** (*bool*) – Whether posts with this resource should be protected.
Context: edit
- **public** (*bool*) – Whether posts of this resource should be shown in the front end of the site.
Context: view, edit
- **queryable** (*bool*) – Whether posts with this resource should be publicly-queryable.
Context: view, edit
- **show_in_list** (*bool*) – Whether to include posts in the edit listing for their post type.
Context: edit
- **slug** (*str*) – An alphanumeric identifier for the resource.
Context: embed, view, edit

classmethod parse (*api, json*)

class `wordpress.models.PostType` (*api=None*)

Bases: `wordpress.models.Model`

class `wordpress.models.ResultSet`

Bases: `list`

A list of like object that holds results from the WordPress API query.

ids ()

class `wordpress.models.Setting` (*api=None*)

Bases: `wordpress.models.Model`

```
class wordpress.models.Tag(api=None)
```

Bases: `wordpress.models.Model`

A WordPress post object.

Parameters

- **id** (*int*) – Unique identifier for the term.
Context: view, embed, edit
- **count** (*int*) – Number of published posts for the term.
Context: view, edit
- **description** (*str*) – HTML description of the term.
Context: view, edit
- **link** (*str*) – URL of the term.
Context: view, embed, edit
- **name** (*str*) – HTML title for the term.
Context: view, embed, edit
- **slug** (*str*) – An alphanumeric identifier for the term unique to its type.
Context: view, embed, edit
- **taxonomy** (*str*) – Type attribution for the term.
Context: view, embed, edit
One of: category, post_tag, nav_menu, link_category, post_format
- **meta** (*dict*) – Meta fields.
Context: view, edit

```
delete (**kwargs)
```

```
classmethod parse(api, json)
```

```
update (**kwargs)
```

```
class wordpress.models.Taxonomy(api=None)
```

Bases: `wordpress.models.Model`

```
class wordpress.models.User(api=None)
```

Bases: `wordpress.models.Model`

wordpress.utils module

```
wordpress.utils.parse_iso8601(string)
```

Module contents

```
class wordpress.WordPress(url, verify_ssl=True)
```

Bases: `object`

```
create_comment (**kwargs)
```

```
create_media (**kwargs)
```


create_page (***kwargs*)

create_post (*date=None, date_gmt=None, slug=None, status=None, password=None, title=None, content=None, author=None, excerpt=None, featured_media=None, comment_status=None, ping_status=None, format=None, meta=None, sticky=None, template=None, categories=None, tags=None, liveblog_links=None*)

Create a Post.

Parameters

- **date** (*datetime*) – The date the object was published, in the site’s timezone.
- **date_gmt** (*datetime*) – The date the object was published, as GMT.
- **slug** (*str*) – An alphanumeric identifier for the object unique to its type.
- **status** (*str*) – A named status for the object.
One of: publish, future, draft, pending, private
- **password** (*str*) – A password to protect access to the content and excerpt.
- **title** (*str*) – The title for the object.
- **content** (*str*) – The content for the object.
- **author** (*id*) – The ID for the author of the object.
- **excerpt** (*str*) – The excerpt for the object.
- **featured_media** (*int*) – The ID of the featured media for the object.
- **comment_status** (*str*) – Whether or not comments are open on the object.
One of: open, closed
- **ping_status** (*str*) – Whether or not the object can be pinged.
One of: open, closed
- **format** (*str*) – The format for the object.
One of: standard
- **meta** (*dict*) – Meta fields.
- **sticky** (*bool*) – Whether or not the object should be treated as sticky.
- **template** (*str*) – The theme file to use to display the object.
One of:
- **categories** (*str*) – The terms assigned to the object in the category taxonomy.
- **tags** (*str*) – The terms assigned to the object in the post_tag taxonomy.
- **liveblog_likes** (*str*) – The number of Liveblog Likes the post has.

create_tag (***kwargs*)

create_user (***kwargs*)

delete_comment (***kwargs*)

delete_media (***kwargs*)

delete_page (***kwargs*)

delete_post (*pk, force=False*)

Delete a Post.

Parameters

- **pk** (*int*) – The post id you want to delete.
- **force** (*bool*) – Whether to bypass trash and force deletion.

delete_post_revision (*parent, pk*)

Delete Post Revision.

Parameters

- **parent** (*int/wordpress.models.Post/wordpress.models.Page*) – The id for the parent of the object.
- **pk** (*int*) – Unique identifier for the object.

Returns

Return type dict

delete_tag (***kwargs*)

delete_user (***kwargs*)

get_category (*pk, context='view'*)

Retrieve a Category.

Parameters

- **pk** (*int*) – The category id you want to retrieve.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

Returns

Return type *wordpress.models.Category*

get_comment (***kwargs*)

get_media (***kwargs*)

get_page (***kwargs*)

get_post (*pk, context='view', password=None*)

Retrieve a Post.

Parameters

- **pk** (*int*) – The post id you want to retrieve.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

- **password** (*str*) – The password for the post if it is password protected.

Returns

Return type *wordpress.models.Post*

get_post_revision (*parent, pk, context='view'*)

Get a Post Revision.

Parameters

- **parent** (*int/wordpress.models.Post/wordpress.models.Page*) – The id for the parent of the object.
- **pk** (*int*) – Unique identifier for the object.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view

Returns

Return type *wordpress.models.PostRevision*

get_post_status (*slug, context='view'*)

Retrieve a Post statuses

Parameters

- **slug** (*str*) – The name of the status.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

Returns

Return type *wordpress.models.PostStatus*

get_post_type (***kwargs*)

get_tag (*pk, context='view'*)

Retrieve a Tag.

Parameters

- **pk** (*int*) – The tag id you want to retrieve.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

Returns

Return type *wordpress.models.Tag*

get_taxonomy (***kwargs*)

get_user (***kwargs*)

list_categories (*context='view', page=1, pre_page=10, search=None, exclude=None, include=None, order='asc', orderby='name', hide_empty=False, parent=None, post=None, slug=None*)

Get a list of categories.

Parameters

- **context** (*str*) – Scope under which the request is made; determines fields present in response.
Default: view
One of: view, embed, edit
- **page** (*int*) – Current page of the collection.
Default: 1
- **pre_page** (*int*) – Maximum number of items to be returned in result set.
Default: 10
- **search** (*str*) – Limit results to those matching a string.
- **exclude** (*int*) – Ensure result set excludes specific IDs.
Default:
- **include** (*int*) – Limit result set to specific IDs.
Default:
- **order** (*str*) – Order sort attribute ascending or descending.
Default: asc
One of: asc, desc
- **orderby** (*str*) – Sort collection by term attribute.
Default: name
One of: id, include, name, slug, term_group, description, count
- **hide_empty** (*bool*) – Whether to hide terms not assigned to any posts.
- **parent** (*int/wordpress.models.Category*) – Limit result set to terms assigned to a specific parent.
- **post** (*int/wordpress.models.Post*) – Limit result set to terms assigned to a specific post.
- **slug** (*str*) – Limit result set to terms with a specific slug.

Returns A list of `wordpress.models.Category`.

Return type list

list_comments (***kwargs*)

list_media (***kwargs*)

list_pages (***kwargs*)

list_post_revisions (*parent, context='view'*)

List Post Revisions.

Parameters

- **parent** (*int/wordpress.models.Post/wordpress.models.Page*) – The id for the parent of the object.
- **context** (*str*) – Scope under which the request is made; determines fields present in response.
Default: view

One of: view

Returns A list of `wordpress.models.PostRevision`.

Return type list

list_post_statuses (*context*='view')

Get a list of post statuses.

Parameters **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

Returns A list of `wordpress.models.PostStatus`

Return type list

list_post_types (***kwargs*)

list_posts (*context*='view', *page*=1, *pre_page*=10, *search*=None, *after*=None, *author*=None, *author_exclude*=None, *before*=None, *exclude*=None, *include*=None, *offset*=None, *order*='desc', *orderby*='date', *slug*=None, *status*='publish', *categories*=None, *categories_exclude*=None, *tags*=None, *tags_exclude*=None, *sticky*=None)

Get a list of posts.

Parameters

- **context** (*str*) – Scope under which the request is made; determines fields present in response.

Default: view

One of: view, embed, edit

- **page** (*int*) – Current page of the collection.
 - **pre_page** (*int*) – Maximum number of items to be returned in result set.
 - **search** (*str*) – Limit results to those matching a string.
 - **after** (*datetime*) – Limit response to posts published after a given date.
 - **author** (*int*) – Limit result set to posts assigned to specific authors.
 - **author_exclude** (*int*) – Ensure result set excludes posts assigned to specific authors.
 - **before** (*datetime*) – Limit response to posts published before a given date.
 - **exclude** (*int*) – Ensure result set excludes specific IDs.
 - **include** (*int*) – Limit result set to specific IDs.
 - **offset** (*int*) – Offset the result set by a specific number of items.
 - **order** (*str*) – Order sort attribute ascending or descending.
- Default: desc
- One of: asc, desc
- **orderby** (*str*) – Sort collection by object attribute. Default: date One of: date, relevance, id, include, title, slug
 - **slug** (*str*) – Limit result set to posts with one or more specific slugs.

- **status** (*str*) – Limit result set to posts assigned one or more statuses.
Default: publish
One of: publish, future, draft, pending, private
- **categories** (*str*) – Limit result set to all items that have the specified term assigned in the categories taxonomy.
- **categories_exclude** (*str*) – Limit result set to all items except those that have the specified term assigned in the categories taxonomy.
- **tags** (*str*) – Limit result set to all items that have the specified term assigned in the tags taxonomy.
- **tags_exclude** (*str*) – Limit result set to all items except those that have the specified term assigned in the tags taxonomy.
- **sticky** (*bool*) – Limit result set to items that are sticky.

Returns A list of `wordpress.models.Post`.

Return type list

list_tags (*context='view', page=1, pre_page=10, search=None, include=[], offset=0, order='asc', orderby='name', hide_empty=False, post=None, slug=None*)

Get a list of tags.

Parameters

- **context** (*str*) – Scope under which the request is made; determines fields present in response.
Default: view
One of: view, embed, edit
- **page** (*int*) – Current page of the collection.
Default: 1
- **per_page** (*int*) – Maximum number of items to be returned in result set.
Default: 10
- **search** (*str*) – Limit results to those matching a string.
- **exclude** (*str*) – Ensure result set excludes specific IDs.
Default:
- **include** (*list*) – Limit result set to specific IDs.
Default:
- **offset** (*int*) – Offset the result set by a specific number of items.
- **order** (*str*) – Order sort attribute ascending or descending.
Default: asc
One of: asc, desc
- **orderby** (*str*) – Sort collection by term attribute.
Default: name
One of: id, include, name, slug, term_group, description, count
- **hide_empty** (*bool*) – Whether to hide terms not assigned to any posts.

- **post** (*int*) – Limit result set to terms assigned to a specific post.
- **slug** (*str*) – Limit result set to terms with a specific slug.

Returns A list of `wordpress.models.Category`.

Return type `list`

`list_taxonomies (**kwargs)`

`list_users (**kwargs)`

`update_comment (**kwargs)`

`update_media (**kwargs)`

`update_page (**kwargs)`

`update_post` (*pk*, *date=None*, *date_gmt=None*, *slug=None*, *status=None*, *password=None*, *title=None*, *content=None*, *author=None*, *excerpt=None*, *featured_media=None*, *comment_status=None*, *ping_status=None*, *format=None*, *meta=None*, *sticky=None*, *template=None*, *categories=None*, *tags=None*, *liveblog_links=None*)

Update a Post.

Parameters

- **pk** (*int*) – The ID of the post you want to update.
- **date** (*datetime*) – The date the object was published, in the site’s timezone.
- **date_gmt** (*datetime*) – The date the object was published, as GMT.
- **slug** (*str*) – An alphanumeric identifier for the object unique to its type.
- **status** (*str*) – A named status for the object.

One of: publish, future, draft, pending, private

- **password** (*str*) – A password to protect access to the content and excerpt.
- **title** (*str*) – The title for the object.
- **content** (*str*) – The content for the object.
- **author** (*id*) – The ID for the author of the object.
- **excerpt** (*str*) – The excerpt for the object.
- **featured_media** (*int*) – The ID of the featured media for the object.
- **comment_status** (*str*) – Whether or not comments are open on the object.

One of: open, closed

- **ping_status** (*str*) – Whether or not the object can be pinged.

One of: open, closed

- **format** (*str*) – The format for the object.

One of: standard

- **meta** (*dict*) – Meta fields.
- **sticky** (*bool*) – Whether or not the object should be treated as sticky.
- **template** (*str*) – The theme file to use to display the object.

One of:

- **categories** (*str*) – The terms assigned to the object in the category taxonomy.

- **tags** (*str*) – The terms assigned to the object in the post_tag taxonomy.
- **liveblog_likes** (*str*) – The number of Liveblog Likes the post has.

update_setting (*title=None, description=None, url=None, email=None, timezone=None, date_format=None, time_format=None, start_of_week=None, language=None, use_smilies=None, default_category=None, default_post_format=None, post_pre_page=None*)

Update WordPress settings.

Parameters

- **title** (*str*) – Site title.
- **description** (*str*) – Site description.
- **url** (*str*) – Site URL.
- **email** (*str*) – This address is used for admin purposes. If you change this we will send you an email at your new address to confirm it. The new address will not become active until confirmed.
- **timezone** (*str*) – A city in the same timezone as you.
- **date_format** (*str*) – A date format for all date strings.
- **time_format** (*str*) – A time format for all time strings.
- **start_of_week** (*int*) – A day number of the week that the week should start on.
- **language** (*str*) – WordPress locale code.
- **use_smilies** (*bool*) – Convert emoticons like :-) and :-P to graphics on display.
- **default_category** (*int*) – Default category.
- **default_post_format** (*str*) – Default post format.
- **posts_per_page** (*int*) – Blog pages show at most.

update_tag (***kwargs*)

update_user (***kwargs*)

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

You can contribute in many ways:

Types of Contributions

Report Bugs

Report bugs at <https://github.com/myles/python-wp/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” and “help wanted” is open to whoever wants to implement it.

Implement Features

Look through the GitHub issues for features. Anything tagged with “enhancement” and “help wanted” is open to whoever wants to implement it.

Write Documentation

Python WordPress could always use more documentation, whether as part of the official Python WordPress docs, in docstrings, or even on the web in blog posts, articles, and such.

Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/myles/python-wp/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

Get Started!

Ready to contribute? Here’s how to set up *python-wp* for local development.

1. Fork the *python-wp* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/python-wp.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv python-wp
$ cd python-wp/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you’re done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 python-wp tests
$ python setup.py test or py.test
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .  
$ git commit -m "Your detailed description of your changes."  
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.6, 2.7, 3.3, 3.4 and 3.5, and for PyPy. Check https://travis-ci.org/myles/python-wp/pull_requests and make sure that the tests pass for all supported Python versions.

Tips

To run a subset of tests:

```
$ python -m unittest tests.test_python-wp
```

Credits

Development Lead

- Myles Braithwaite <me@mylesbraithwaite.com>

Contributors

None yet. Why not be the first?

History

0.1.0 (25 January 2016)

Started working on the library.

Indices and tables

- [genindex](#)
- [modindex](#)

- [search](#)

W

`wordpress`, [20](#)
`wordpress.api`, [8](#)
`wordpress.models`, [16](#)
`wordpress.utils`, [20](#)

C

Category (class in wordpress.models), 16
 Comment (class in wordpress.models), 17
 create_comment() (wordpress.api.WordPress method), 8
 create_comment() (wordpress.WordPress method), 20
 create_media() (wordpress.api.WordPress method), 8
 create_media() (wordpress.WordPress method), 20
 create_page() (wordpress.api.WordPress method), 8
 create_page() (wordpress.WordPress method), 20
 create_post() (wordpress.api.WordPress method), 8
 create_post() (wordpress.WordPress method), 21
 create_tag() (wordpress.api.WordPress method), 9
 create_tag() (wordpress.WordPress method), 21
 create_user() (wordpress.api.WordPress method), 9
 create_user() (wordpress.WordPress method), 21

D

delete() (wordpress.models.Category method), 17
 delete() (wordpress.models.Post method), 18
 delete() (wordpress.models.Tag method), 20
 delete_comment() (wordpress.api.WordPress method), 9
 delete_comment() (wordpress.WordPress method), 21
 delete_media() (wordpress.api.WordPress method), 9
 delete_media() (wordpress.WordPress method), 21
 delete_page() (wordpress.api.WordPress method), 9
 delete_page() (wordpress.WordPress method), 21
 delete_post() (wordpress.api.WordPress method), 9
 delete_post() (wordpress.WordPress method), 21
 delete_post_revision() (wordpress.api.WordPress method), 9
 delete_post_revision() (wordpress.WordPress method), 22
 delete_tag() (wordpress.api.WordPress method), 9
 delete_tag() (wordpress.WordPress method), 22
 delete_user() (wordpress.api.WordPress method), 9
 delete_user() (wordpress.WordPress method), 22
 destroy() (wordpress.models.PostRevision method), 19

G

get_category() (wordpress.api.WordPress method), 9
 get_category() (wordpress.WordPress method), 22
 get_comment() (wordpress.api.WordPress method), 10
 get_comment() (wordpress.WordPress method), 22
 get_media() (wordpress.api.WordPress method), 10
 get_media() (wordpress.WordPress method), 22
 get_page() (wordpress.api.WordPress method), 10
 get_page() (wordpress.WordPress method), 22
 get_post() (wordpress.api.WordPress method), 10
 get_post() (wordpress.WordPress method), 22
 get_post_revision() (wordpress.api.WordPress method), 10
 get_post_revision() (wordpress.WordPress method), 22
 get_post_status() (wordpress.api.WordPress method), 10
 get_post_status() (wordpress.WordPress method), 23
 get_post_type() (wordpress.api.WordPress method), 11
 get_post_type() (wordpress.WordPress method), 23
 get_tag() (wordpress.api.WordPress method), 11
 get_tag() (wordpress.WordPress method), 23
 get_taxonomy() (wordpress.api.WordPress method), 11
 get_taxonomy() (wordpress.WordPress method), 23
 get_user() (wordpress.api.WordPress method), 11
 get_user() (wordpress.WordPress method), 23

I

ids() (wordpress.models.ResultSet method), 19

L

list_categories() (wordpress.api.WordPress method), 11
 list_categories() (wordpress.WordPress method), 23
 list_comments() (wordpress.api.WordPress method), 12
 list_comments() (wordpress.WordPress method), 24
 list_media() (wordpress.api.WordPress method), 12
 list_media() (wordpress.WordPress method), 24
 list_pages() (wordpress.api.WordPress method), 12
 list_pages() (wordpress.WordPress method), 24
 list_post_revisions() (wordpress.api.WordPress method), 12

`list_post_revisions()` (wordpress.WordPress method), 24
`list_post_statuses()` (wordpress.api.WordPress method), 12
`list_post_statuses()` (wordpress.WordPress method), 25
`list_post_types()` (wordpress.api.WordPress method), 12
`list_post_types()` (wordpress.WordPress method), 25
`list_posts()` (wordpress.api.WordPress method), 12
`list_posts()` (wordpress.WordPress method), 25
`list_tags()` (wordpress.api.WordPress method), 14
`list_tags()` (wordpress.WordPress method), 26
`list_taxonomies()` (wordpress.api.WordPress method), 14
`list_taxonomies()` (wordpress.WordPress method), 27
`list_users()` (wordpress.api.WordPress method), 14
`list_users()` (wordpress.WordPress method), 27

M

Media (class in `wordpress.models`), 17
Model (class in `wordpress.models`), 17

P

Page (class in `wordpress.models`), 17
`parse()` (wordpress.models.Category class method), 17
`parse()` (wordpress.models.Model class method), 17
`parse()` (wordpress.models.Post class method), 18
`parse()` (wordpress.models.PostRevision class method), 19
`parse()` (wordpress.models.PostStatus class method), 19
`parse()` (wordpress.models.Tag class method), 20
`parse_iso8601()` (in module `wordpress.utils`), 20
`parse_list()` (wordpress.models.Model class method), 17
Post (class in `wordpress.models`), 17
PostRevision (class in `wordpress.models`), 18
PostStatus (class in `wordpress.models`), 19
PostType (class in `wordpress.models`), 19

R

ResultSet (class in `wordpress.models`), 19
`revision()` (wordpress.models.Post method), 18
`revisions()` (wordpress.models.Post method), 18

S

Setting (class in `wordpress.models`), 19

T

Tag (class in `wordpress.models`), 19
Taxonomy (class in `wordpress.models`), 20

U

`update()` (wordpress.models.Category method), 17
`update()` (wordpress.models.Post method), 18
`update()` (wordpress.models.Tag method), 20
`update_comment()` (wordpress.api.WordPress method), 14

`update_comment()` (wordpress.WordPress method), 27
`update_media()` (wordpress.api.WordPress method), 14
`update_media()` (wordpress.WordPress method), 27
`update_page()` (wordpress.api.WordPress method), 14
`update_page()` (wordpress.WordPress method), 27
`update_post()` (wordpress.api.WordPress method), 14
`update_post()` (wordpress.WordPress method), 27
`update_setting()` (wordpress.api.WordPress method), 15
`update_setting()` (wordpress.WordPress method), 28
`update_tag()` (wordpress.api.WordPress method), 16
`update_tag()` (wordpress.WordPress method), 28
`update_user()` (wordpress.api.WordPress method), 16
`update_user()` (wordpress.WordPress method), 28
User (class in `wordpress.models`), 20

W

WordPress (class in `wordpress`), 20
WordPress (class in `wordpress.api`), 8
`wordpress` (module), 20
`wordpress.api` (module), 8
`wordpress.models` (module), 16
`wordpress.utils` (module), 20