
pip-helpers Documentation

Release 0.6.post22.dev210748605

Christian Fobel

Jul 20, 2017

Contents

1	pip-helpers	1
2	Credits	3
2.1	Project Modules	3
3	Indices and tables	5
	Python Module Index	7

CHAPTER 1

pip-helpers

Functions to perform common operations with the Python Package Index.

Example usage:

```
>>> import pip_helpers as ph
>>>
>>> # Install list of Python packages
>>> stdout_and_stderr = ph.install(['numpy>=1.11.1', 'pandas<0.18.1'])
>>>
>>> # Uninstall list of installed Python packages
>>> stdout_and_stderr = ph.uninstall(['numpy', 'pandas'])
>>>
>>> # Get list of descriptors for installed Python packages
>>> installed_package_descriptors = ph.freeze()
```


Written by Christian Fobel christian@fobel.net

Contents:

Project Modules

pip_helpers Package

pip_helpers Package

`pip_helpers.freeze()`

Returns Sorted list of package descriptors (e.g., "foo", "foo==1.0", "foo>=1.0"), one descriptor for each installed package.

Return type list

`pip_helpers.get_releases(package_str, pre=False, key=None, include_hidden=False, server_url='https://pypi.python.org/pypi/{}/json', hidden_url=None)`

Query Python Package Index for list of available release for specified package.

Parameters

- **package_str** (*str*) – Name of package hosted on Python package index. Version constraints are also supported (e.g., "foo", "foo==1.0", "foo>=1.0", etc.). See [version specifiers](#) reference for more details.
- **pre** (*bool*, *optional*) – Include pre-release packages.
- **key** (*function*, *optional*) – Key function to sort (package_name, release_info) items by.
- **include_hidden** (*bool*, *optional*) – Include “hidden” packages.

- **server_url** (*str*, *optional*) – URL to JSON API (default=“<https://pypi.python.org/pypi/{}/json>”).
- **hidden_url** (*str*, *optional*) – URL to XMLRPC API (default=“<https://pypi.python.org/pypi/>” for PyPI server URL).

Returns Package name and package release information, indexed by package version string and ordered by upload time (i.e., most recent release is last).

Return type (*string*, *collections.OrderedDict*)

`pip_helpers.install(packages, capture_streams=True)`

Install the specified list of packages from the Python Package Index.

Parameters

- **packages** (*list*) – List of package descriptors (e.g., “foo”, “foo==1.0”, “foo>=1.0”).
- **capture_streams** (*bool*, *optional*) – If True, capture stdout and stderr output and instead print concise progress indicator.

Returns Combined output to stdout and stderr.

Return type *str*

`pip_helpers.uninstall(packages, capture_streams=True)`

Uninstall the specified list of Python packages

Parameters

- **packages** (*list*) – List of package names (e.g., “foo”, but not “foo==1.0”, “foo>=1.0”).
- **capture_streams** (*bool*, *optional*) – If True, capture stdout and stderr output and instead print concise progress indicator.

Returns Combined output to stdout and stderr.

Return type *str*

`pip_helpers.upgrade(package_name)`

Upgrade package, without upgrading dependencies that are already satisfied.

See [here](#) for more details.

Parameters **package_name** (*str*) – Package name.

Returns

Dictionary containing:

- **original_version**: Package version before upgrade.
- **new_version**: Package version after upgrade (*None* if package was not upgraded).
- **installed_dependencies**: List of dependencies installed during package upgrade. Each dependency is represented as a dictionary of the form {‘package’: ..., ‘version’: ...}.

Return type *dict*

Raises `pkg_resources.DistributionNotFound` – If package not installed.

CHAPTER 3

Indices and tables

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

p

`pip_helpers`, 3

F

`freeze()` (in module `pip_helpers`), 3

G

`get_releases()` (in module `pip_helpers`), 3

I

`install()` (in module `pip_helpers`), 4

P

`pip_helpers` (module), 3

U

`uninstall()` (in module `pip_helpers`), 4

`upgrade()` (in module `pip_helpers`), 4