pypi-simple

Release 0.9.0

John T. Wodder II

2021 Aug 26

CONTENTS

1 High-Level API **3**

2 Low-Level Utilities **9**
- 2.1 Parsing Simple Repository Pages . . . 9
- 2.2 Parsing Filenames . . . 13

3 Changelog **15**
- 3.1 v0.9.0 (2021-08-26) . . . 15
- 3.2 v0.8.0 (2020-12-13) . . . 15
- 3.3 v0.7.0 (2020-10-15) . . . 15
- 3.4 v0.6.0 (2020-03-01) . . . 16
- 3.5 v0.5.0 (2019-05-12) . . . 16
- 3.6 v0.4.0 (2018-09-06) . . . 16
- 3.7 v0.3.0 (2018-09-03) . . . 17
- 3.8 v0.2.0 (2018-09-01) . . . 17
- 3.9 v0.1.0 (2018-08-31) . . . 17

4 Installation **19**

5 Example **21**

6 Indices and tables **23**

Python Module Index **25**

Index **27**

GitHub | PyPI | Documentation | Issues | *Changelog*

CHAPTER

ONE

HIGH-LEVEL API

class pypi_simple.**PyPISimple**(*endpoint: str = 'https://pypi.org/simple/', auth: Optional[Any] = None, session: Optional[requests.sessions.Session] = None*)

A client for fetching package information from a Python simple package repository.

If necessary, login/authentication details for the repository can be specified at initialization by setting the `auth` parameter to either a `(username, password)` pair or another authentication object accepted by requests.

If more complicated session configuration is desired (e.g., setting up caching), the user must create & configure a `requests.Session` object appropriately and pass it to the constructor as the `session` parameter.

A *`PyPISimple`* instance can be used as a context manager that will automatically close its session on exit, regardless of where the session object came from.

Changed in version 0.8.0: Now usable as a context manager

Changed in version 0.5.0: `session` argument added

Changed in version 0.4.0: `auth` argument added

Parameters

- **endpoint** (*str*) – The base URL of the simple API instance to query; defaults to the base URL for PyPI's simple API
- **auth** – Optional login/authentication details for the repository; either a `(username, password)` pair or another authentication object accepted by requests
- **session** – Optional `requests.Session` object to use instead of creating a fresh one

get_index_page(*timeout: Optional[Union[float, Tuple[float, float]]] = None*) → pypi_simple.classes.IndexPage

New in version 0.7.0.

Fetches the index/root page from the simple repository and returns an *`IndexPage`* instance.

> **Warning:** PyPI's project index file is very large and takes several seconds to parse. Use this method sparingly.

Parameters **timeout** (*Union[float, Tuple[float,float], None]*) – optional timeout to pass to the `requests` call

Return type *IndexPage*

Raises

- **requests.HTTPError** – if the repository responds with an HTTP error code

- **UnsupportedRepoVersionError** – if the repository version has a greater major component than the supported repository version

stream_project_names(*chunk_size: int = 65535, timeout: Optional[Union[float, Tuple[float, float]]] = None*) → Iterator[str]

New in version 0.7.0.

Returns a generator of names of projects available in the repository. The names are not normalized.

Unlike `get_index_page()` and `get_projects()`, this function makes a streaming request to the server and parses the document in chunks. It is intended to be faster than the other methods, especially when the complete document is very large.

> **Warning:** This function is rather experimental. It does not have full support for web encodings, encoding detection, or handling invalid HTML.

Parameters

- **chunk_size** (*int*) – how many bytes to read from the response at a time
- **timeout** (*Union[float, Tuple[float,float], None]*) – optional timeout to pass to the `requests` call

Return type Iterator[str]

Raises

- **requests.HTTPError** – if the repository responds with an HTTP error code
- **UnsupportedRepoVersionError** – if the repository version has a greater major component than the supported repository version

get_project_page(*project: str, timeout: Optional[Union[float, Tuple[float, float]]] = None*) → Optional[pypi_simple.classes.ProjectPage]

New in version 0.7.0.

Fetches the page for the given project from the simple repository and returns a `ProjectPage` instance. Returns `None` if the repository responds with a 404. All other HTTP errors cause a `requests.HTTPError` to be raised.

Parameters

- **project** (*str*) – The name of the project to fetch information on. The name does not need to be normalized.
- **timeout** (*Union[float, Tuple[float,float], None]*) – optional timeout to pass to the `requests` call

Return type Optional[*ProjectPage*]

Raises

- **requests.HTTPError** – if the repository responds with an HTTP error code other than 404
- **UnsupportedRepoVersionError** – if the repository version has a greater major component than the supported repository version

get_project_url(*project: str*) → str

Returns the URL for the given project's page in the repository.

Parameters **project** (*str*) – The name of the project to build a URL for. The name does not need to be normalized.

Return type str

get_projects() → Iterator[str]

Returns a generator of names of projects available in the repository. The names are not normalized.

> **Warning:** PyPI's project index file is very large and takes several seconds to parse. Use this method sparingly.

Deprecated since version 0.7.0: Use *get_index_page()* or *stream_project_names()* instead

Return type Iterator[str]

Raises

- **requests.HTTPError** – if the repository responds with an HTTP error code
- ***UnsupportedRepoVersionError*** – if the repository version has a greater major component than the supported repository version

get_project_files(*project: str*) → List[pypi_simple.classes.DistributionPackage]

Returns a list of *DistributionPackage* objects representing all of the package files available in the repository for the given project.

When fetching the project's information from the repository, a 404 response is treated the same as an empty page, resulting in an empty list. All other HTTP errors cause a `requests.HTTPError` to be raised.

Deprecated since version 0.7.0: Use *get_project_page()* instead

Parameters **project** (*str*) – The name of the project to fetch information on. The name does not need to be normalized.

Return type List[*DistributionPackage*]

Raises

- **requests.HTTPError** – if the repository responds with an HTTP error code other than 404
- ***UnsupportedRepoVersionError*** – if the repository version has a greater major component than the supported repository version

class pypi_simple.**IndexPage**(*projects: List[str], repository_version: Optional[str], last_serial: Optional[str]*)

New in version 0.7.0.

A parsed index/root page from a simple repository

property projects

The project names listed in the index. The names are not normalized.

property repository_version

The repository version reported by the page, or `None` if not specified

property last_serial

The value of the *X-PyPI-Last-Serial* response header returned when fetching the page, or `None` if not specified

class pypi_simple.**ProjectPage**(*project: str, packages: List[pypi_simple.classes.DistributionPackage], repository_version: Optional[str], last_serial: Optional[str]*)

New in version 0.7.0.

A parsed project page from a simple repository

property project
The name of the project the page is for

property packages
A list of packages (as `DistributionPackage` objects) listed on the project page

property repository_version
The repository version reported by the page, or `None` if not specified

property last_serial
The value of the `X-PyPI-Last-Serial` response header returned when fetching the page, or `None` if not specified

class pypi_simple.**DistributionPackage**(*filename: str, url: str, project: Optional[str], version: Optional[str], package_type: Optional[str], requires_python: Optional[str], has_sig: Optional[bool], yanked: Optional[str], metadata_digests: Optional[Dict[str, str]]*)

Information about a versioned archive file from which a Python project release can be installed

Changed in version 0.5.0: `yanked` attribute added

Changed in version 0.9.0: `has_metadata`, `metadata_url`, and `metadata_digests` attributes added

property filename
The basename of the package file

property url
The URL from which the package file can be downloaded

property project
The name of the project (as extracted from the filename), or `None` if the filename cannot be parsed

property version
The project version (as extracted from the filename), or `None` if the filename cannot be parsed

property package_type
The type of the package, or `None` if the filename cannot be parsed. The recognized package types are:

- `'dumb'`
- `'egg'`
- `'msi'`
- `'rpm'`
- `'sdist'`
- `'wheel'`
- `'wininst'`

property requires_python
An optional version specifier string declaring the Python version(s) in which the package can be installed

property has_sig
Whether the package file is accompanied by a PGP signature file. This is `None` if the package repository does not report such information.

Changed in version 0.7.0: Will now be `None` if not specified by repository; previously would be `False` in such a situation

property yanked
If the package file has been "yanked" from the package repository (meaning that it should only be installed when that specific version is requested), this attribute will be a string giving the reason why it was yanked; otherwise, it is `None`.

property metadata_digests
If the package repository provides a Core Metadata file for the package, this is a (possibly empty) `dict` of digests of the file, given as a mapping from hash algorithm names to hex-encoded digest strings; otherwise, it is `None`

property sig_url
The URL of the package file's PGP signature file, if it exists; cf. *has_sig*

Changed in version 0.6.0: Now always defined; would previously be `None` if *has_sig* was false

property has_metadata
Whether the package file is accompanied by a Core Metadata file

property metadata_url
If the package repository provides a Core Metadata file for the package, this is the URL for that file; otherwise, it is `None`.

get_digests() → Dict[str, str]
Extracts the hash digests from the package file's URL and returns a `dict` mapping hash algorithm names to hex-encoded digest strings

classmethod from_link(*link: pypi_simple.classes.Link*, *project_hint: Optional[str] = None*) → pypi_simple.classes.DistributionPackage
New in version 0.7.0.

Construct a *DistributionPackage* from a *Link* on a project page.

Parameters

- **link** (*Link*) – a link parsed from a project page
- **project_hint** (*Optional[str]*) – Optionally, the expected value for the project name (usually the name of the project page on which the link was found). The name does not need to be normalized.

Return type *DistributionPackage*

pypi_simple.**PYPI_SIMPLE_ENDPOINT: str = 'https://pypi.org/simple/'**
The base URL for PyPI's simple API

pypi_simple.**SUPPORTED_REPOSITORY_VERSION: str = '1.0'**
The maximum supported simple repository version (See **PEP 629**)

exception pypi_simple.**UnsupportedRepoVersionError**(*declared_version: str*, *supported_version: str*)
Raised upon encountering a simple repository whose repository version (**PEP 629**) has a greater major component than the maximum supported repository version (*SUPPORTED_REPOSITORY_VERSION*)

declared_version: str
The version of the simple repository

supported_version: str
The maximum repository version that we support

CHAPTER

TWO

LOW-LEVEL UTILITIES

2.1 Parsing Simple Repository Pages

`pypi_simple.parse_repo_index_page`(*html: Union[str, bytes], from_encoding: Optional[str] = None*) → pypi_simple.classes.IndexPage

New in version 0.7.0.

Parse an index/root page from a simple repository into an `IndexPage`. Note that the `last_serial` attribute will be `None`.

Parameters

- **html** (`str or bytes`) – the HTML to parse
- **from_encoding** (`Optional[str]`) – an optional hint to Beautiful Soup as to the encoding of `html` when it is `bytes` (usually the `charset` parameter of the response's `Content-Type` header)

Return type *IndexPage*

Raises **`UnsupportedRepoVersionError`** – if the repository version has a greater major component than the supported repository version

`pypi_simple.parse_repo_index_response`(*r: requests.models.Response*) → pypi_simple.classes.IndexPage

New in version 0.7.0.

Parse an index page from a `requests.Response` returned from a (non-streaming) request to a simple repository, and return an `IndexPage`.

Parameters **r** (`requests.Response`) – the response object to parse

Return type *IndexPage*

Raises **`UnsupportedRepoVersionError`** – if the repository version has a greater major component than the supported repository version

`pypi_simple.parse_repo_project_page`(*project: str, html: Union[str, bytes], base_url: Optional[str] = None, from_encoding: Optional[str] = None*) → pypi_simple.classes.ProjectPage

New in version 0.7.0.

Parse a project page from a simple repository into a `ProjectPage`. Note that the `last_serial` attribute will be `None`.

Parameters

- **project** (`str`) – The name of the project whose page is being parsed
- **html** (`str or bytes`) – the HTML to parse

- **base_url** (*Optional[str]*) – an optional URL to join to the front of the packages' URLs (usually the URL of the page being parsed)
- **from_encoding** (*Optional[str]*) – an optional hint to Beautiful Soup as to the encoding of `html` when it is `bytes` (usually the `charset` parameter of the response's *Content-Type* header)

Return type *ProjectPage*

Raises ***UnsupportedRepoVersionError*** – if the repository version has a greater major component than the supported repository version

`pypi_simple.`**`parse_repo_project_response`**(*project: str, r: requests.models.Response*) → pypi_simple.classes.ProjectPage

New in version 0.7.0.

Parse a project page from a `requests.Response` returned from a (non-streaming) request to a simple repository, and return a *ProjectPage*.

Parameters

- **project** (*str*) – The name of the project whose page is being parsed
- **r** (*requests.Response*) – the response object to parse

Return type *ProjectPage*

Raises ***UnsupportedRepoVersionError*** – if the repository version has a greater major component than the supported repository version

`pypi_simple.`**`parse_repo_links`**(*html: Union[str, bytes], base_url: Optional[str] = None, from_encoding: Optional[str] = None*) → Tuple[Dict[str, str], List[pypi_simple.classes.Link]]

New in version 0.7.0.

Parse an HTML page from a simple repository and return a `(metadata, links)` pair.

The `metadata` element is a `Dict[str, str]`. Currently, the only key that may appear in it is `"repository_version"`, which maps to the repository version reported by the HTML page in accordance with **PEP 629**. If the HTML page does not contain a repository version, this key is absent from the `dict`.

The `links` element is a list of *Link* objects giving the hyperlinks found in the HTML page.

Parameters

- **html** (*str or bytes*) – the HTML to parse
- **base_url** (*Optional[str]*) – an optional URL to join to the front of the links' URLs (usually the URL of the page being parsed)
- **from_encoding** (*Optional[str]*) – an optional hint to Beautiful Soup as to the encoding of `html` when it is `bytes` (usually the `charset` parameter of the response's *Content-Type* header)

Return type Tuple[Dict[str, str], List[*Link*]]

Raises ***UnsupportedRepoVersionError*** – if the repository version has a greater major component than the supported repository version

class `pypi_simple.`**`Link`**(*text: str, url: str, attrs: Dict[str, Union[str, List[str]]]*)

New in version 0.7.0.

A hyperlink extracted from an HTML page

property text
The text inside the link tag, with leading & trailing whitespace removed and with any tags nested inside the link tags ignored

property url
The URL that the link points to, resolved relative to the URL of the source HTML page and relative to the page's `<base>` href value, if any

property attrs
A dictionary of attributes set on the link tag (including the unmodified `href` attribute). Keys are converted to lowercase. Most attributes have `str` values, but some (referred to as "CDATA list attributes" by the HTML spec; e.g., `"class"`) have values of type `List[str]` instead.

2.1.1 Streaming Parsers

`pypi_simple.`**`parse_links_stream`**(*htmlseq: Iterable*, *base_url: Optional[str] = None*, *http_charset: Optional[str] = None*) → Iterator[pypi_simple.classes.Link]

New in version 0.7.0.

Parse an HTML page given as an iterable of `bytes` or `str` and yield each hyperlink encountered in the document as a *`Link`* object.

This function consumes the elements of `htmlseq` one at a time and yields the links found in each segment before moving on to the next one. It is intended to be faster than both *`parse_links()`* and *`parse_repo_links()`*, especially when the complete document is very large.

> **Warning:** This function is rather experimental. It does not have full support for web encodings, encoding detection, or handling invalid HTML. It also leaves CDATA list attributes on links as strings instead of converting them to lists.

Parameters

- **`htmlseq`** (*`Iterable[AnyStr]`*) – an iterable of either `bytes` or `str` that, when joined together, form an HTML document to parse
- **`base_url`** (*`Optional[str]`*) – an optional URL to join to the front of the links' URLs (usually the URL of the page being parsed)
- **`http_charset`** (*`Optional[str]`*) – the document's encoding as declared by the transport layer, if any; e.g., as declared in the `charset` parameter of the *`Content-Type`* header of the HTTP response that returned the document

Return type Iterator[*Link*]

Raises ***`UnsupportedRepoVersionError`*** – if the repository version has a greater major component than the supported repository version

`pypi_simple.`**`parse_links_stream_response`**(*r: requests.models.Response*, *chunk_size: int = 65535*) → Iterator[pypi_simple.classes.Link]

New in version 0.7.0.

Parse an HTML page from a streaming `requests.Response` object and yield each hyperlink encountered in the document as a *`Link`* object.

See *`parse_links_stream()`* for more information.

Parameters

- **r** (*requests.Response*) – the streaming response object to parse
- **chunk_size** (*int*) – how many bytes to read from the response at a time

Return type Iterator[*Link*]

Raises ***UnsupportedRepoVersionError*** – if the repository version has a greater major component than the supported repository version

2.1.2 Deprecated Functions

pypi_simple.**parse_simple_index**(*html: Union[str, bytes], base_url: Optional[str] = None, from_encoding: Optional[str] = None*) → Iterator[Tuple[str, str]]

Parse a simple repository's index page and return a generator of `(project name, project URL)` pairs

Deprecated since version 0.7.0: Use *parse_repo_index_page()* or *parse_links_stream()* instead

Parameters

- **html** (*str or bytes*) – the HTML to parse
- **base_url** (*Optional[str]*) – an optional URL to join to the front of the URLs returned (usually the URL of the page being parsed)
- **from_encoding** (*Optional[str]*) – an optional hint to Beautiful Soup as to the encoding of `html` when it is `bytes` (usually the `charset` parameter of the response's *Content-Type* header)

Return type Iterator[Tuple[str, str]]

Raises ***UnsupportedRepoVersionError*** – if the repository version has a greater major component than the supported repository version

pypi_simple.**parse_project_page**(*html: Union[str, bytes], base_url: Optional[str] = None, from_encoding: Optional[str] = None, project_hint: Optional[str] = None*) → List[pypi_simple.classes.DistributionPackage]

Parse a project page from a simple repository and return a list of *DistributionPackage* objects

Deprecated since version 0.7.0: Use *parse_repo_project_page()* instead

Parameters

- **html** (*str or bytes*) – the HTML to parse
- **base_url** (*Optional[str]*) – an optional URL to join to the front of the packages' URLs (usually the URL of the page being parsed)
- **from_encoding** (*Optional[str]*) – an optional hint to Beautiful Soup as to the encoding of `html` when it is `bytes` (usually the `charset` parameter of the response's *Content-Type* header)
- **project_hint** (*Optional[str]*) – The name of the project whose page is being parsed; used to disambiguate the parsing of certain filenames

Return type List[*DistributionPackage*]

Raises ***UnsupportedRepoVersionError*** – if the repository version has a greater major component than the supported repository version

`pypi_simple.parse_links`(*html: Union[str, bytes]*, *base_url: Optional[str] = None*, *from_encoding: Optional[str] = None*) → Iterator[Tuple[str, str, Dict[str, Union[str, List[str]]]]]

Parse an HTML page and return a generator of links, where each link is represented as a triple of link text, link URL, and a `dict` of link tag attributes (including the unmodified `href` attribute).

Link text has all leading & trailing whitespace removed.

Keys in the attributes `dict` are converted to lowercase.

Deprecated since version 0.7.0: Use *`parse_repo_links()`* instead

Parameters

- **`html`** (*`str or bytes`*) – the HTML to parse
- **`base_url`** (*`Optional[str]`*) – an optional URL to join to the front of the URLs returned (usually the URL of the page being parsed)
- **`from_encoding`** (*`Optional[str]`*) – an optional hint to Beautiful Soup as to the encoding of `html` when it is `bytes` (usually the `charset` parameter of the response's *`Content-Type`* header)

Return type Iterator[Tuple[str, str, Dict[str, Union[str, List[str]]]]]

2.2 Parsing Filenames

`pypi_simple.parse_filename`(*filename: str*, *project_hint: Optional[str] = None*) → Union[Tuple[str, str, str], Tuple[None, None, None]]

Given the filename of a distribution package, returns a triple of the project name, project version, and package type. The name and version are spelled the same as they appear in the filename; no normalization is performed.

The package type may be any of the following strings:

- `'dumb'`
- `'egg'`
- `'msi'`
- `'rpm'`
- `'sdist'`
- `'wheel'`
- `'wininst'`

If the filename cannot be parsed, `(None, None, None)` is returned.

Note that some filenames (e.g., `1-2-3.tar.gz`) may be ambiguous as to which part is the project name and which is the version. In order to resolve the ambiguity, the expected value for the project name (*modulo* normalization) can be supplied as the `project_name` argument to the function. If the filename can be parsed with the given string in the role of the project name, the results of that parse will be returned; otherwise, the function will fall back to breaking the project & version apart at an unspecified point.

Parameters

- **`filename`** (*`str`*) – The package filename to parse
- **`project_hint`** (*`Optional[str]`*) – Optionally, the expected value for the project name (usually the name of the project page on which the filename was found). The name does not need to be normalized.

Return type Union[Tuple[str, str, str], Tuple[None, None, None]]

CHAPTER

THREE

CHANGELOG

3.1 v0.9.0 (2021-08-26)

- Support **PEP 658** by adding `has_metadata`, `metadata_url`, and `metadata_digests` attributes to `DistributionPackage`

3.2 v0.8.0 (2020-12-13)

- Support Python 3.9
- `PyPISimple` is now usable as a context manager that will close the session on exit

3.3 v0.7.0 (2020-10-15)

- Drop support for Python 2.7, Python 3.4, and Python 3.5
- `DistributionPackage.has_sig` is now `None` if the package repository does not report this information
- Added type annotations
- Moved documentation from README file to a Read the Docs site
- Added new methods to `PyPISimple`:
  - `get_index_page()` — Returns an `IndexPage` instance with a `projects: List[str]` attribute plus other attributes for repository metadata
  - `get_project_page()` — Returns a `ProjectPage` instance with a `packages: List[DistributionPackage]` attribute plus other attributes for repository metadata
  - `stream_project_names()` — Retrieves project names from a repository using a streaming request
- New utility functions:
  - `parse_repo_links()` — Parses an HTML page and returns a pair of repository metadata and a list of `Link` objects
  - `parse_repo_project_page()` — Parses a project page and returns a `ProjectPage` instance
  - `parse_repo_project_response()` — Parses a `requests.Response` object containing a project page and returns a `ProjectPage` instance
  - `parse_links_stream()` — Parses an HTML page as stream of `bytes` or `str` and returns a generator of `Link` objects

  - *parse_links_stream_response()* — Parses a streaming `requests.Response` object containing an HTML page and returns a generator of *Link* objects
  - *parse_repo_index_page()* — Parses a simple repository index/root page and returns an *IndexPage* instance
  - *parse_repo_index_response()* — Parses a `requests.Response` object containing an index page and returns an *IndexPage* instance
- The following functions & methods are now deprecated and will be removed in a future version:
  - *PyPISimple.get_projects()*
  - *PyPISimple.get_project_files()*
  - *parse_simple_index()*
  - *parse_project_page()*
  - *parse_links()*
- Support Warehouse's *X-PyPI-Last-Serial* header by attaching the value to the objects returned by *get_index_page()* and *get_project_page()*
- Support **PEP 629** by attaching the repository version to the objects returned by *get_index_page()* and *get_project_page()* and by raising an *UnsupportedRepoVersionError* when a repository with an unsupported version is encountered

3.4 v0.6.0 (2020-03-01)

- Support Python 3.8
- *DistributionPackage.sig_url* is now always non-`None`, as Warehouse does not report proper values for *has_sig*

3.5 v0.5.0 (2019-05-12)

- The *PyPISimple* constructor now takes an optional `session` argument which can be used to specify a `requests.Session` object with more complicated configuration than just authentication
- Support for PEP 592; *DistributionPackage* now has a *yanked* attribute

3.6 v0.4.0 (2018-09-06)

- Publicly (i.e., in the README) document the utility functions
- Gave *PyPISimple* an `auth` parameter for specifying login/authentication details

3.7 v0.3.0 (2018-09-03)

- When fetching the list of files for a project, the project name is now used to resolve ambiguous filenames.
- The filename parser now requires all filenames to be all-ASCII (except for wheels).

3.8 v0.2.0 (2018-09-01)

- The filename parser now rejects invalid project names, blatantly invalid versions, and non-ASCII digits.
- RPM packages are now recognized.

3.9 v0.1.0 (2018-08-31)

Initial release

`pypi-simple` is a client library for the Python Simple Repository API as specified in **PEP 503** and updated by **PEP 592**, **PEP 629**, and **PEP 658**. With it, you can query the Python Package Index (PyPI) and other pip-compatible repositories for a list of their available projects and lists of each project's available package files. The library also allows you to query package files for their project version, package type, file digests, `requires_python` string, PGP signature URL, and metadata URL.

CHAPTER

FOUR

INSTALLATION

`pypi-simple` requires Python 3.6 or higher. Just use pip for Python 3 (You have pip, right?) to install `pypi-simple` and its dependencies:

```
python3 -m pip install pypi-simple
```

CHAPTER

FIVE

EXAMPLE

```
>>> from pypi_simple import PyPISimple
>>> with PyPISimple() as client:
...     requests_page = client.get_project_page('requests')
>>> pkg = requests_page.packages[0]
>>> pkg
DistributionPackage(filename='requests-0.2.0.tar.gz', url='https://files.pythonhosted.
↪org/packages/ba/bb/dfa0141a32d773c47e4dede1a617c59a23b74dd302e449cf85413fc96bc4/
↪requests-0.2.0.tar.gz
↪#sha256=813202ace4d9301a3c00740c700e012fb9f3f8c73ddcfe02ab558a8df6f175fd', project=
↪'requests', version='0.2.0', package_type='sdist', requires_python=None, has_
↪sig=None, yanked=None, metadata_digests=None)
>>> pkg.filename
'requests-0.2.0.tar.gz'
>>> pkg.url
'https://files.pythonhosted.org/packages/ba/bb/
↪dfa0141a32d773c47e4dede1a617c59a23b74dd302e449cf85413fc96bc4/requests-0.2.0.tar.gz
↪#sha256=813202ace4d9301a3c00740c700e012fb9f3f8c73ddcfe02ab558a8df6f175fd'
>>> pkg.project
'requests'
>>> pkg.version
'0.2.0'
>>> pkg.package_type
'sdist'
>>> pkg.get_digests()
{'sha256': '813202ace4d9301a3c00740c700e012fb9f3f8c73ddcfe02ab558a8df6f175fd'}
```

CHAPTER

SIX

INDICES AND TABLES

- genindex
- search

PYTHON MODULE INDEX

p

pypi_simple, 1

INDEX

A

attrs() (*pypi_simple.Link property*), 11

D

declared_version (*pypi_simple.UnsupportedRepoVersionError attribute*), 7
DistributionPackage (*class in pypi_simple*), 6

F

filename() (*pypi_simple.DistributionPackage property*), 6
from_link() (*pypi_simple.DistributionPackage class method*), 7

G

get_digests() (*pypi_simple.DistributionPackage method*), 7
get_index_page() (*pypi_simple.PyPISimple method*), 3
get_project_files() (*pypi_simple.PyPISimple method*), 5
get_project_page() (*pypi_simple.PyPISimple method*), 4
get_project_url() (*pypi_simple.PyPISimple method*), 4
get_projects() (*pypi_simple.PyPISimple method*), 5

H

has_metadata() (*pypi_simple.DistributionPackage property*), 7
has_sig() (*pypi_simple.DistributionPackage property*), 6

I

IndexPage (*class in pypi_simple*), 5

L

last_serial() (*pypi_simple.IndexPage property*), 5
last_serial() (*pypi_simple.ProjectPage property*), 6
Link (*class in pypi_simple*), 10

M

metadata_digests() (*pypi_simple.DistributionPackage property*), 7
metadata_url() (*pypi_simple.DistributionPackage property*), 7
module
  pypi_simple, 1

P

package_type() (*pypi_simple.DistributionPackage property*), 6
packages() (*pypi_simple.ProjectPage property*), 6
parse_filename() (*in module pypi_simple*), 13
parse_links() (*in module pypi_simple*), 12
parse_links_stream() (*in module pypi_simple*), 11
parse_links_stream_response() (*in module pypi_simple*), 11
parse_project_page() (*in module pypi_simple*), 12
parse_repo_index_page() (*in module pypi_simple*), 9
parse_repo_index_response() (*in module pypi_simple*), 9
parse_repo_links() (*in module pypi_simple*), 10
parse_repo_project_page() (*in module pypi_simple*), 9
parse_repo_project_response() (*in module pypi_simple*), 10
parse_simple_index() (*in module pypi_simple*), 12
project() (*pypi_simple.DistributionPackage property*), 6
project() (*pypi_simple.ProjectPage property*), 6
ProjectPage (*class in pypi_simple*), 5
projects() (*pypi_simple.IndexPage property*), 5
pypi_simple
  module, 1
PYPI_SIMPLE_ENDPOINT (*in module pypi_simple*), 7
PyPISimple (*class in pypi_simple*), 3

Python Enhancement Proposals
PEP 503, 17
PEP 592, 17
PEP 629, 7, 10, 16, 17
PEP 658, 15, 17

R

repository_version() (*pypi_simple.IndexPage property*), 5
repository_version() (*pypi_simple.ProjectPage property*), 6
requires_python() (*pypi_simple.DistributionPackage property*), 6

S

sig_url() (*pypi_simple.DistributionPackage property*), 7
stream_project_names() (*pypi_simple.PyPISimple method*), 4
SUPPORTED_REPOSITORY_VERSION (*in module pypi_simple*), 7
supported_version (*pypi_simple.UnsupportedRepoVersionError attribute*), 7

T

text() (*pypi_simple.Link property*), 10

U

UnsupportedRepoVersionError, 7
url() (*pypi_simple.DistributionPackage property*), 6
url() (*pypi_simple.Link property*), 11

V

version() (*pypi_simple.DistributionPackage property*), 6

Y

yanked() (*pypi_simple.DistributionPackage property*), 6