

---

# **pyLDAPAPI Client Documentation**

**CSIRO Land and Water**

**Nov 30, 2018**



---

## Getting Started

---

<b>1</b>	<b>Welcome to pyLDAPAPI Client</b>	<b>1</b>
	<b>Python Module Index</b>	<b>5</b>



# CHAPTER 1

---

## Welcome to pyLDAPAPI Client

---

The Python client library for the Python Linked Data API (pyLDAPAPI) is:

*A Simple helper library for consuming registers, indexes, and instances of classes exposed via a pyLDAPAPI endpoint.*

See the [pyLDAPAPI](#) module for more information.

PyPI Badge - *coming soon*

## 1.1 Example usage

```
from pyldapi_client import LDAPIClient

async def async_test_script(loop):
    remapper = {
        "http://test.linked.data.gov.au/dataset/asgs/": "http://13.236.122.60/asgs/
↪sa4/"
    }
    base = "http://test.linked.data.gov.au/dataset/asgs/reg"
    client = await LDAPIClient(base, url_remapper=remapper, asynchronous=True,
↪loop=loop)
    register = client.register("http://test.linked.data.gov.au/dataset/asgs/sa4/")
    first_page = await register.index_page(per_page=50)
    a = await register.index()
    instances = await register.instances(index=a, min_count=20)
    _ = await client.close()
    print(len(a))
    print(len(instances))
    return

def threaded_test_script():
    remapper = {
        "http://test.linked.data.gov.au/dataset/asgs/": "http://13.236.122.60/asgs/
↪sa4/"
```

(continues on next page)

(continued from previous page)

```

    }
    base = "http://test.linked.data.gov.au/dataset/asgs/reg"
    client = LDAPIClient(base, url_remapper=remapper, asynchronous=False, threads=8)
    register = client.register("http://test.linked.data.gov.au/dataset/asgs/sa4/")
    first_page = register.index_page(per_page=50)
    a = register.index()
    instances = register.instances(index=a, min_count=20)
    client.close()
    print(len(a))
    print(len(instances))
    return

def sync_test_script():
    remapper = {
        "http://test.linked.data.gov.au/dataset/asgs/": "http://13.236.122.60/asgs/
↪sa4/"
    }
    base = "http://test.linked.data.gov.au/dataset/asgs/reg"
    client = LDAPIClient(base, url_remapper=remapper, asynchronous=False, threads=1)
    register = client.register("http://test.linked.data.gov.au/dataset/asgs/sa4/")
    first_page = register.index_page(per_page=50)
    a = register.index()
    instances = register.instances(index=a, min_count=20)
    client.close()
    print(len(a))
    print(len(instances))
    return

if __name__ == "__main__":
    import asyncio
    # For debugging/testing
    threaded_test_script()
    sync_test_script()
    loop = asyncio.get_event_loop()
    _a = asyncio_test_script(loop)
    loop.run_until_complete(_a)

```

## 1.2 Documentation

Read the documentation at <http://pyldapi-client.readthedocs.io/>.

## 1.3 Implementations of pyLDAPAPI Client

- LOC-I Index File Exporter

## 1.4 Licence

This module is licensed under Apache Software License v2.0. See the [LICENSE deed](#) for details.

## 1.5 Contact

### 1.5.1 Nicholas Car (lead)

*Senior Experimental Scientist*  
CSIRO Land and Water  
nicholas.car@csiro.au  
<http://orcid.org/0000-0002-8742-7730>

### 1.5.2 Ashley Sommer (senior developer)

*Informatics Software Engineer*  
CSIRO Land and Water  
ashley.sommer@csiro.au

## Installation

*Coming soon.*

## Setup

*Coming soon.*

## Usage

*Coming soon.*

## Indices and tables

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

## Utility functions

Utility functions shared between the py27 and py35 versions of pyLDAPAPI Client. In theory, all of these functions should be compatible with *all* versions of python.

```
pyldapi_client.functions.extract_page_from_string(page_string)
```

TODO: Ashley

**Parameters** `page_string` –

**Returns**

**Return type**

```
pyldapi_client.functions.extract_per_page_from_string(page_string)
```

TODO: Ashley

**Parameters** `page_string` –

**Returns**

**Return type**

`pyldapi_client.functions.find_registers_from_ld_payload(uri, json_payload, container)`

TODO: Ashley

**Parameters**

- `uri` –
- `json_payload` –
- `container` –

**Returns**

**Return type**

## LDAPIClient

*This is the documentation for python3+ however, the API should be the same for the python2.7 version as well.*

**class** `pyldapi_client.py35.LDAPIClient` (`base_uri`, `*args`, `url_remapper=None`, `threads=1`, `**kwargs`)

Bases: `pyldapi_client.py35.AbstractLDAPIClient`

TODO: Ashley

**base\_uri**

**close()**

**register** (`reg_uri`)

TODO: Ashley

**Parameters** `reg_uri` –

**Returns**

**Return type**

**session**

**threads**

**url\_remapper**



### p

`pyldapi_client.functions`, 3  
`pyldapi_client.py35`, 4



## B

`base_uri` (`pyldapi_client.py35.LDAPIClient` attribute), 4

## C

`close()` (`pyldapi_client.py35.LDAPIClient` method), 4

## E

`extract_page_from_string()` (in module `pyldapi_client.functions`), 3

`extract_per_page_from_string()` (in module `pyldapi_client.functions`), 3

## F

`find_registers_from_ld_payload()` (in module `pyldapi_client.functions`), 4

## L

`LDAPIClient` (class in `pyldapi_client.py35`), 4

## P

`pyldapi_client.functions` (module), 3

`pyldapi_client.py35` (module), 4

## R

`register()` (`pyldapi_client.py35.LDAPIClient` method), 4

## S

`session` (`pyldapi_client.py35.LDAPIClient` attribute), 4

## T

`threads` (`pyldapi_client.py35.LDAPIClient` attribute), 4

## U

`url_remapper` (`pyldapi_client.py35.LDAPIClient` attribute), 4