
Barique

unknown

Sep 23, 2021

CONTENTS

1	Barique	3
1.1	Installation	3
1.2	Examples	3
1.3	License	4
2	Commands	5
2.1	file	5
2.2	task	7
3	baricadr package	9
3.1	Subpackages	9
3.2	Submodules	11
3.3	Module contents	11
4	Indices and tables	13
	Python Module Index	15
	Index	17

Contents:

BARIQUE

Lint [Documentation](#)

A Python client (library and CLI) for interacting with a [Baricadr](#) server.

1.1 Installation

```
$ pip install barique

# On first use you'll need to create a config file to connect to the server, just run:

$ barique init
Welcome to Barique
Baricadr server host: localhost
Baricadr server port: 9100
Is your Baricadr instance running behind an authentication proxy? [y/N]: n
Testing connection...
Ok! Everything looks good.
Ready to go! Type `barique` to get a list of commands you can execute.
```

This will create a barique config file in `~/.barique.yml`

1.2 Examples

```
# List files in a remote repository
$ barique file list /repos/test_repo/
[
  {
    "Path": "file.txt"
  },
  {
    "Path": "file2.txt"
  }
]

# Ask Baricadr to pull a single file
$ barique file pull /repos/test_repo/file.txt
7958b29c-2a14-486c-90f0-585e68ac9f44
```

(continues on next page)

(continued from previous page)

```
# Check the status of a pull task
$ barique task show 7958b29c-2a14-486c-90f0-585e68ac9f44
{
  "created": "Wed, 17 Feb 2021 15:11:56 GMT",
  "error": null,
  "finished": "Wed, 17 Feb 2021 15:12:03 GMT",
  "path": "/repos/test_repo/file.txt",
  "started": "Wed, 17 Feb 2021 15:11:58 GMT",
  "status": "finished",
  "task_id": "7958b29c-2a14-486c-90f0-585e68ac9f44",
  "type": "pull"
}

# Ask Baricadr to freeze a single file
$ barique file freeze /repos/test_repo/file2.txt
7958b29c-2a14-486c-90f0-585e68ac9f44

# Check the status of a pull task
$ barique task show 7958b29c-2a14-486c-90f0-585e68ac9f44
{
  "created": "Wed, 17 Feb 2021 15:16:49 GMT",
  "error": null,
  "finished": "Wed, 17 Feb 2021 15:16:52 GMT",
  "path": "/repos/test_repo/file2.txt",
  "started": "Wed, 17 Feb 2021 15:16:51 GMT",
  "status": "finished",
  "task_id": "7958b29c-2a14-486c-90f0-585e68ac9f44",
  "type": "freeze"
}
```

1.3 License

Available under the MIT License

COMMANDS

Barique is a set of wrappers for Baricadr's API. It builds a set of small, useful utilities for talking to a Baricadr server. Each utility is implemented as a subcommand of `barique`. This section of the documentation describes these commands.

2.1 file

This section is auto-generated from the help text for the `barique` command `file`.

2.1.1 freeze command

Usage:

```
barique file freeze [OPTIONS] PATH
```

Help

Launch a freeze task

Output

Id associated to the freeze task

Options:

```
--force      Force freezing, even if the freezing delay was not reached
--dry_run    Do not make any deletion, just list changes that would be made
--email TEXT User email adress for notification
-h, --help   Show this message and exit.
```

2.1.2 list command

Usage:

```
barique file list [OPTIONS] PATH
```

Help

List files available from a remote repository for a local path

Output

List of file relative paths

Options:

<code>--full</code>	List full information for each file (size (in bytes), mtime, etc..)
<code>--missing</code>	Only list files missing from the local path
<code>--max_depth INTEGER</code>	Restrict to a max depth. Set to 0 for all files. [default: 1]
<code>--from_root</code>	Return full paths from root of the repo (instead of relative to given path)
<code>-h, --help</code>	Show this message and exit.

2.1.3 pull command

Usage:

```
barique file pull [OPTIONS] PATH
```

Help

Launch a pull task

Output

Id associated to the pull task

Options:

<code>--email TEXT</code>	User email adress for notification
<code>--dry_run</code>	Do not make any pull, just list changes that would be made
<code>-h, --help</code>	Show this message and exit.

2.1.4 tree command

Usage:

```
barique file tree [OPTIONS] PATH
```

Help

List files available from a remote repository for a local path as a tree

Output

None

Options:

<code>--max_depth INTEGER</code>	Restrict to a max depth. Set to 0 for all files.
<code>-h, --help</code>	Show this message and exit.

2.2 task

This section is auto-generated from the help text for the barique command `task`.

2.2.1 list command

Usage:

```
barique task list [OPTIONS]
```

Help

List recent tasks

Output

List of tasks

Options:

```
-h, --help Show this message and exit.
```

2.2.2 log command

Usage:

```
barique task log [OPTIONS] TASK_ID
```

Help

Show log from the task with the specified id

Output

Dict

Options:

```
-h, --help Show this message and exit.
```

2.2.3 remove command

Usage:

```
barique task remove [OPTIONS] TASK_ID
```

Help

Remove task with the selected id

Output

Dict

Options:

```
-h, --help  Show this message and exit.
```

2.2.4 show command

Usage:

```
barique task show [OPTIONS] TASK_ID
```

Help

Show task with the specified id

Output

Dict

Options:

```
-h, --help  Show this message and exit.
```

2.2.5 zombies command

Usage:

```
barique task zombies [OPTIONS]
```

Help

Kill zombie tasks

Output

Task id

Options:

```
-h, --help  Show this message and exit.
```

BARICADR PACKAGE

3.1 Subpackages

3.1.1 baricadr.file package

Module contents

class `baricadr.file.FileClient(url, login, password, endpoints)`

Bases: `baricadr.client.Client`

Manipulate files managed by Baricadr

freeze(*path*, *force=False*, *dry_run=False*, *email=""*)

Launch a freeze task

Parameters

- **path** (*str*) – Local path to a file or folder to freeze
- **force** (*bool*) – Force freezing, even if the freezing delay was not reached
- **dry_run** (*bool*) – Do not make any deletion, just list changes that would be made
- **email** (*str*) – User email adress for notification

Return type `str`

Returns Id associated to the freeze task

list(*path*, *full=False*, *missing=False*, *max_depth=1*, *from_root=False*)

List files available from a remote repository for a local path

Parameters

- **path** (*str*) – Local path
- **missing** (*bool*) – Only list files missing from the local path
- **full** (*bool*) – List full information for each file (size (in bytes), mtime, etc..)
- **max_depth** (*int*) – Restrict to a max depth. Set to 0 for all files.
- **from_root** (*bool*) – Return full paths from root of the repo (instead of relative to given path)

Return type `list`

Returns List of file relative paths

pull(*path*, *email*="", *dry_run*=False)

Launch a pull task

Parameters

- **path** (*str*) – Local path to a missing file or folder
- **email** (*str*) – User email adress for notification
- **dry_run** (*bool*) – Do not make any pull, just list changes that would be made

Return type *str*

Returns Id associated to the pull task

tree(*path*, *max_depth*=0)

List files available from a remote repository for a local path as a tree

Parameters

- **path** (*str*) – Local path
- **max_depth** (*int*) – Restrict to a max depth. Set to 0 for all files.

Return type *None*

Returns *None*

3.1.2 baricadr.task package

Module contents

class `baricadr.task.TaskClient`(*url*, *login*, *password*, *endpoints*)

Bases: `baricadr.client.Client`

Track progress of Baricadr tasks

list()

List recent tasks

Return type *list*

Returns List of tasks

log(*task_id*)

Show log from the task with the specified id

Parameters **task_id** (*str*) – Task id

Return type *dict*

Returns *Dict*

remove(*task_id*)

Remove task with the selected id

Parameters **task_id** (*str*) – Task id

Return type *dict*

Returns *Dict*

show(*task_id*)

Show task with the specified id

Parameters **task_id** (*str*) – Task id

Return type dict

Returns Dict

zombies()

Kill zombie tasks

Return type dict

Returns Task id

3.2 Submodules

3.2.1 baricadr.client module

class baricadr.client.**Client**(*url, login, password, endpoints*)

Bases: object

Base client class implementing methods to make queries to the server

3.2.2 baricadr.exceptions module

exception baricadr.exceptions.**BaricadrApiError**

Bases: Exception

Raised when the API returns an error

exception baricadr.exceptions.**BaricadrConnectionError**

Bases: Exception

Raised when the connection to the Barricadr server fails

exception baricadr.exceptions.**BaricadrNotImplementedError**

Bases: Exception

Raised when there are not endpoint associated with this function

3.3 Module contents

class baricadr.**BaricadrInstance**(*url='http://localhost:9100/', login=None, password=None, **kwargs*)

Bases: object

INDICES AND TABLES

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

PYTHON MODULE INDEX

b

- `baricadr`, [11](#)
- `baricadr.client`, [11](#)
- `baricadr.exceptions`, [11](#)
- `baricadr.file`, [9](#)
- `baricadr.task`, [10](#)

INDEX

B

baricadr
 module, 11
baricadr.client
 module, 11
baricadr.exceptions
 module, 11
baricadr.file
 module, 9
baricadr.task
 module, 10
BaricadrApiError, 11
BaricadrConnectionError, 11
BaricadrInstance (class in baricadr), 11
BaricadrNotImplementedError, 11

C

Client (class in baricadr.client), 11

F

FileClient (class in baricadr.file), 9
freeze() (baricadr.file.FileClient method), 9

L

list() (baricadr.file.FileClient method), 9
list() (baricadr.task.TaskClient method), 10
log() (baricadr.task.TaskClient method), 10

M

module
 baricadr, 11
 baricadr.client, 11
 baricadr.exceptions, 11
 baricadr.file, 9
 baricadr.task, 10

P

pull() (baricadr.file.FileClient method), 9

R

remove() (baricadr.task.TaskClient method), 10

S

show() (baricadr.task.TaskClient method), 10

T

TaskClient (class in baricadr.task), 10
tree() (baricadr.file.FileClient method), 10

Z

zombies() (baricadr.task.TaskClient method), 11