
open-humans-api Documentation

Open Humans

Sep 04, 2019

Contents:

1	Installation	3
1.1	Command Line Tools	3
1.2	ohapi and submodules	11
1.3	Development & Testing	20
2	Indices and tables	21
	Python Module Index	23
	Index	25

`open-humans-api` is a Python package that wraps the API methods of [Open Humans](#) for easier use in your own Python applications and websites.

It also installs a set of command line utilities that can be used to

- download public files
 - for a given Project
 - for a given Member
- Upload files for a project through a `master_access_token` or `OAuth2 access_token`
- Download files for a project through a `master_access_token` or `OAuth2 access_token`

You can install `open-humans-api` through `pip`:

```
pip install open-humans-api
```

1.1 Command Line Tools

1.1.1 Example Use Cases

Download public files for a given project/user

You can use `ohpub-download` on the command line to download all public files associated with a project. Here is an example that will download all public 23andMe files:

```
mkdir 23andme # create new folder to store the data in
ohpub-download --source direct-sharing-128 --directory 23andme
```

To do the same for all public files of a given user:

```
mkdir gedankenstuecke # create new folder to store the data in
ohpub-download --username gedankenstuecke --directory gedankenstuecke
```

Download private files from members of your project

If you are running a project yourself you can use a `master_access_token` you can get from the [Open Humans website](#) to download the files shared with you.

Here is an example of how to do this:

```
mkdir my_downloaded_data # create new folder to store the data in
ohproj-download --master-token my_master_access_token --directory my_downloaded_data
```

Upload files to the accounts of your project members

Uploading data into your members accounts is two-step procedure using both `ohproj-upload-metadata` as well as `ohproj-upload`, as you first need to draft some metadata that will go along with the files.

Both commands expect a directory that contains sub-directories named after your members `project_member_id`. An example directory structure with files could look like this:

- `member_data/`
 - `01234567/`
 - * `testdata.json`
 - * `testdata.txt`
 - `12345678/`
 - * `testdata.json`
 - * `testdata.txt`

To upload this data we need to have a CSV file that contains the metadata. We can draft one using the following command:

```
ohproj-upload-metadata -d member_data --create-csv member_data_metadata.csv
```

The resulting CSV will look like this:

```
$ cat member_data_metadata.csv =>
project_member_id,filename,tags,description,md5,creation_date
01234567,testdata.txt,,fa61a92e21a2597900cbde09d8ddbc1a,2016-08-23T15:23:22.
↪277060+00:00
01234567,testdata.json,json,,577da9879649acaf17226a6461bd19c8,2016-08-23T16:06:16.
↪415039+00:00
12345678,testdata.txt,,fa61a92e21a2597900cbde09d8ddbc1a,2016-09-20T10:10:59.
↪863201+00:00
12345678,testdata.json,json,,577da9879649acaf17226a6461bd19c8,2016-09-20T10:10:59.
↪859201+00:00
```

Edit this CSV with a text or spreadsheet editor of your choice to contain the metadata and then save it as a CSV again:

```
$ cat member_data_metadata.csv =>
project_member_id,filename,tags,description,md5,creation_date
1234567,testdata.txt,"txt, verbose-data",Complete test data in text format.,
↪fa61a92e21a2597900cbde09d8ddbc1a,2016-08-23T15:23:22.277060+00:00
1234567,testdata.json,"json, metadata",Summary metadata in JSON format.,
↪577da9879649acaf17226a6461bd19c8,2016-08-23T16:06:16.415039+00:00
12345678,testdata.txt,"txt, verbose-data",Complete test data in text format.,
↪fa61a92e21a2597900cbde09d8ddbc1a,2016-09-20T10:10:59.863201+00:00
12345678,testdata.json,"json, metadata",Summary test data JSON.,
↪577da9879649acaf17226a6461bd19c8,2016-09-20T10:10:59.859201+00:00
```

We filled in the tags as a "-"-escaped and comma-separated list as well as a text-description. With this we can now perform the actual upload like this:

```
ohproj-upload -T YOUR_MASTER_ACCESS_TOKEN --metadata-csv member_data_metadata.csv -d_
↪member_data
```

This will upload the data from the `member_data` directory along with the metadata specified in `member_data_metadata.csv`.

Using an OAuth2 access_token

The CLI tools also accept the usage of an OAuth2 access_token``s instead of an ``master_access_token. This enables you to use the CLI tools in your own OAuth2 applications if you're not using Python for your development.

Here is an example to download all the data shared by a single member through the command line:

```
ohproj-download -t personal_access_token -m '12345678' --directory download_directory
```

1.1.2 List of commands and their parameters

ohpub-download

Command line tools for downloading public data.

```
ohpub-download [OPTIONS]
```

Options

- s, --source** <source>
the source to download files from
- u, --username** <username>
the user to download files from
- d, --directory** <directory>
the directory for downloaded files
- m, --max-size** <max_size>
the maximum file size to download
- q, --quiet**
Report ERROR level logging to stdout
- debug**
Report DEBUG level logging to stdout.

ohproj-download

Command line function for downloading data from project members to the target directory. For more information visit [download](#).

```
ohproj-download [OPTIONS]
```

Options

- d, --directory** <directory>
Target directory for downloaded files. [required]
- T, --master-token** <master_token>
Project master access token.

- m, --member** <member>
Project member ID.
- t, --access-token** <access_token>
OAuth2 user access token.
- s, --source** <source>
Only download files from this source.
- project-data**
Download this project's own data.
- max-size** <max_size>
Maximum file size to download. [default: 128m]
- v, --verbose**
Report INFO level logging to stdout
- debug**
Report DEBUG level logging to stdout.
- memberlist** <memberlist>
Text file with whitelist IDs to retrieve
- excludelist** <excludelist>
Text file with blacklist IDs to avoid
- id-filename**
Prepend filenames with IDs to ensure uniqueness.

ohproj-download-metadata

Command line function for downloading metadata. For more information visit [download_metadata](#).

```
ohproj-download-metadata [OPTIONS]
```

Options

- T, --master-token** <master_token>
Project master access token. [required]
- v, --verbose**
Show INFO level logging
- debug**
Show DEBUG level logging.
- output-csv** <output_csv>
Output project metadata CSV [required]

ohproj-upload-metadata

Command line function for drafting or reviewing metadata files. For more information visit [upload_metadata](#).

```
ohproj-upload-metadata [OPTIONS]
```

Options

- d, --directory** <directory>
Target directory [required]
- create-csv** <create_csv>
Create draft CSV metadata
- review** <review>
Review existing metadata file
- max-size** <max_size>
Maximum file size to consider. [default: 128m]
- v, --verbose**
Show INFO level logging
- debug**
Show DEBUG level logging.

ohproj-upload

Command line function for uploading files to OH. For more information visit [upload](#).

```
ohproj-upload [OPTIONS]
```

Options

- d, --directory** <directory>
Target directory for downloaded files. [required]
- metadata-csv** <metadata_csv>
CSV file containing file metadata. [required]
- T, --master-token** <master_token>
Project master access token.
- m, --member** <member>
Project member ID.
- t, --access-token** <access_token>
OAuth2 user access token.
- safe**
Do not overwrite files in Open Humans.
- sync**
Delete files not present in local directories.
- max-size** <max_size>
Maximum file size to download. [default: 128m]
- v, --verbose**
Report INFO level logging to stdout
- debug**
Report DEBUG level logging to stdout.

ohproj-oauth2-token-exchange

Command line function for obtaining the refresh token/code. For more information visit [oauth2_token_exchange](#).

```
ohproj-oauth2-token-exchange [OPTIONS]
```

Options

- cid, --client_id** <client_id>
client id of user. [required]
- cs, --client_secret** <client_secret>
client secret of user. [required]
- re_uri, --redirect_uri** <redirect_uri>
redirect_uri of user [required]
- base_url** <base_url>
base url of Open Humans [default: <https://www.openhumans.org/>]
- code** <code>
code of user
- rt, --refresh_token** <refresh_token>
refresh token of user

ohproj-oauth2-url

Command line function for obtaining the Oauth2 url. For more information visit [oauth2_auth_url](#).

```
ohproj-oauth2-url [OPTIONS]
```

Options

- r, --redirect_uri** <redirect_uri>
Redirect URL for project
- c, --client_id** <client_id>
Client ID for project [required]
- base_url** <base_url>
Base URL

ohproj-message

Command line function for sending email to a single user or in bulk. For more information visit [message](#).

```
ohproj-message [OPTIONS]
```

Options

- s, --subject** <subject>
subject [required]
- m, --message_body** <message_body>
compose message [required]
- at, --access_token** <access_token>
access token [required]
- all_members** <all_members>
all members [default: False]
- project_member_ids** <project_member_ids>
list of comma-separated project_member_ids. Example argument: "ID1, ID2"
- v, --verbose**
Show INFO level logging
- debug**
Show DEBUG level logging.

ohproj-delete

Command line function for deleting files. For more information visit [delete_file](#).

```
ohproj-delete [OPTIONS]
```

Options

- T, --access_token** <access_token>
Access token [required]
- m, --project_member_id** <project_member_id>
Project Member ID [required]
- b, --file_basename** <file_basename>
File basename
- i, --file_id** <file_id>
File ID
- all_files** <all_files>
Delete all files [default: False]

`ohapi.command_line.download` (*directory*, *master_token=None*, *member=None*, *access_token=None*, *source=None*, *project_data=False*, *max_size='128m'*, *verbose=False*, *debug=False*, *memberlist=None*, *excludelist=None*, *id_filename=False*)

Download data from project members to the target directory.

Unless this is a member-specific download, directories will be created for each project member ID. Also, unless a source is specified, all shared sources are downloaded and data is sorted into subdirectories according to source.

Projects can optionally return data to Open Humans member accounts. If `project_data` is `True` (or the “`-project-data`” flag is used), this data (the project’s own data files, instead of data from other sources) will be downloaded for each member.

Parameters

- **directory** – This field is the target directory to download data.
- **master_token** – This field is the master access token for the project. It's default value is None.
- **member** – This field is specific member whose project data is downloaded. It's default value is None.
- **access_token** – This field is the user specific access token. It's default value is None.
- **source** – This field is the data source. It's default value is None.
- **project_data** – This field is data related to particular project. It's default value is False.
- **max_size** – This field is the maximum file size. It's default value is 128m.
- **verbose** – This boolean field is the logging level. It's default value is False.
- **debug** – This boolean field is the logging level. It's default value is False.
- **memberlist** – This field is list of members whose data will be downloaded. It's default value is None.
- **excludelist** – This field is list of members whose data will be skipped. It's default value is None.

`ohapi.command_line.download_metadata(master_token, output_csv, verbose=False, debug=False)`

Output CSV with metadata for a project's downloadable files in Open Humans.

Parameters

- **master_token** – This field is the master access token for the project.
- **output_csv** – This field is the target csv file to which metadata is written.
- **verbose** – This boolean field is the logging level. It's default value is False.
- **debug** – This boolean field is the logging level. It's default value is False.

`ohapi.command_line.set_log_level(debug, verbose)`

Function for setting the logging level.

Parameters

- **debug** – This boolean field is the logging level.
- **verbose** – This boolean field is the logging level.

`ohapi.command_line.upload(directory, metadata_csv, master_token=None, member=None, access_token=None, safe=False, sync=False, max_size='128m', mode='default', verbose=False, debug=False)`

Upload files for the project to Open Humans member accounts.

If using a master access token and not specifying member ID:

(1) Files should be organized in subdirectories according to project member ID, e.g.:

```
main_directory/01234567/data.json          main_directory/12345678/data.json
main_directory/23456789/data.json
```

(2) The metadata CSV should have the following format:

1st column: Project member ID 2nd column: filenames 3rd & additional columns: Metadata fields (see below)

If uploading for a specific member: (1) The local directory should not contain subdirectories. (2) The metadata CSV should have the following format: 1st column: filenames 2nd & additional columns: Metadata fields (see below)

The default behavior is to overwrite files with matching filenames on Open Humans, but not otherwise delete files. (Use `-safe` or `-sync` to change this behavior.)

If included, the following metadata columns should be correctly formatted: `'tags'`: should be comma-separated strings `'md5'`: should match the file's md5 hexdigest `'creation_date'`, `'start_date'`, `'end_date'`: ISO 8601 dates or datetimes

Other metadata fields (e.g. `'description'`) can be arbitrary strings. Either specify `sync` as `True` or `safe` as `True` but not both.

Parameters

- **directory** – This field is the target directory from which data will be uploaded.
- **metadata_csv** – This field is the filepath of the metadata csv file.
- **master_token** – This field is the master access token for the project. It's default value is `None`.
- **member** – This field is specific member whose project data is downloaded. It's default value is `None`.
- **access_token** – This field is the user specific access token. It's default value is `None`.
- **safe** – This boolean field will overwrite matching filename. It's default value is `False`.
- **sync** – This boolean field will delete files on Open Humans that are not in the local directory. It's default value is `False`.
- **max_size** – This field is the maximum file size. It's default value is `None`.
- **mode** – This field takes three value `default`, `sync`, `safe`. It's default value is `'default'`.
- **verbose** – This boolean field is the logging level. It's default value is `False`.
- **debug** – This boolean field is the logging level. It's default value is `False`.

```
ohapi.command_line.upload_metadata(directory, create_csv="", review="", max_size='128m',
                                   verbose=False, debug=False)
```

Draft or review metadata files for uploading files to Open Humans. The target directory should either represent files for a single member (no subdirectories), or contain a subdirectory for each project member ID.

Parameters

- **directory** – This field is the directory for which metadata has to be created.
- **create_csv** – This field is the output filepath to which csv file will be written.
- **max_size** – This field is the maximum file size. It's default value is `None`.
- **verbose** – This boolean field is the logging level. It's default value is `False`.
- **debug** – This boolean field is the logging level. It's default value is `False`.

1.2 ohapi and submodules

The `open-humans-api` package will be available in Python as `ohapi`.

In turn `ohapi` contains multiple sub modules to interact with different things like the project & member-based APIs, the public data API and methods for working with files on the local filesystem.

1.2.1 ohapi.api module

Utility functions to use the OAuth2 project API to e.g. message users, download their files, upload new ones and delete existing files.

exception `ohapi.api.SettingsError`

Bases: `exceptions.Exception`

`ohapi.api.delete_file` (*access_token*, *project_member_id=None*, *base_url='https://www.openhumans.org/'*, *file_basename=None*, *file_id=None*, *all_files=False*)

Delete project member files by **file_basename**, **file_id**, or **all_files**. To learn more about Open Humans OAuth2 projects, go to: <https://www.openhumans.org/direct-sharing/oauth2-features/>.

Parameters

- **access_token** – This field is user specific `access_token`.
- **project_member_id** – This field is the project member id of user. It's default value is `None`.
- **base_url** – It is this URL `https://www.openhumans.org`.
- **file_basename** – This field is the name of the file to delete for the particular user for the particular project.
- **file_id** – This field is the id of the file to delete for the particular user for the particular project.
- **all_files** – This is a boolean field to delete all files for the particular user for the particular project.

`ohapi.api.delete_files` (**args*, ***kwargs*)

Alternate name for the `delete_file`.

`ohapi.api.exchange_oauth2_member` (*access_token*, *base_url='https://www.openhumans.org/'*, *all_files=True*)

Returns data for a specific user, including shared data files.

Parameters

- **access_token** – This field is the user specific `access_token`.
- **base_url** – It is this URL `https://www.openhumans.org`.

`ohapi.api.get_all_results` (*starting_page*)

Given starting API query for Open Humans, iterate to get all results.

Parameters `page` (*starting*) – This field is the first page, starting from which results will be obtained.

`ohapi.api.get_page` (*url*)

Get a single page of results.

Parameters `url` – This field is the url from which data will be requested.

`ohapi.api.handle_error` (*r*, *expected_code*)

Helper function to match response of a request to the expected status code

Parameters

- **r** – This field is the response of request.
- **expected_code** – This field is the expected status code for the function.

`ohapi.api.message` (*subject, message, access_token, all_members=False, project_member_ids=None, base_url='https://www.openhumans.org/'*)

Send an email to individual users or in bulk. To learn more about Open Humans OAuth2 projects, go to: <https://www.openhumans.org/direct-sharing/oauth2-features/>

Parameters

- **subject** – This field is the subject of the email.
- **message** – This field is the body of the email.
- **access_token** – This is user specific access token/master token.
- **all_members** – This is a boolean field to send email to all members of the project.
- **project_member_ids** – This field is the list of `project_member_id`.
- **base_url** – It is this URL <https://www.openhumans.org>.

`ohapi.api.oauth2_auth_url` (*redirect_uri=None, client_id=None, base_url='https://www.openhumans.org/'*)

Returns an OAuth2 authorization URL for a project, given Client ID. This function constructs an authorization URL for a user to follow. The user will be redirected to Authorize Open Humans data for our external application. An OAuth2 project on Open Humans is required for this to properly work. To learn more about Open Humans OAuth2 projects, go to: <https://www.openhumans.org/direct-sharing/oauth2-features/>

Parameters

- **redirect_uri** – This field is set to *None* by default. However, if provided, it appends it in the URL returned.
- **client_id** – This field is also set to *None* by default however, is a mandatory field for the final URL to work. It uniquely identifies a given OAuth2 project.
- **base_url** – It is this URL <https://www.openhumans.org>.

`ohapi.api.oauth2_token_exchange` (*client_id, client_secret, redirect_uri, base_url='https://www.openhumans.org/', code=None, refresh_token=None*)

Exchange code or refresh token for a new token and refresh token. For the first time when a project is created, code is required to generate refresh token. Once the refresh token is obtained, it can be used later on for obtaining new access token and refresh token. The user must store the refresh token to obtain the new access token. For more details visit: <https://www.openhumans.org/direct-sharing/oauth2-setup/#setup-oauth2-authorization>

Parameters

- **client_id** – This field is the client id of user.
- **client_secret** – This field is the client secret of user.
- **redirect_uri** – This is the user redirect uri.
- **base_url** – It is this URL <https://www.openhumans.org>
- **code** – This field is used to obtain `access_token` for the first time. It's default value is none.
- **refresh_token** – This field is used to obtain a new `access_token` when the token expires.

`ohapi.api.upload_aws` (*target_filepath, metadata, access_token, base_url='https://www.openhumans.org/', remote_file_info=None, project_member_id=None, max_bytes=128000000*)

Upload a file from a local filepath using the “direct upload” API. Equivalent to `upload_file`. To learn more about this API endpoint see: * <https://www.openhumans.org/direct-sharing/on-site-data-upload/> * <https://www.openhumans.org/direct-sharing/oauth2-data-upload/>

Parameters

- **target_filepath** – This field is the filepath of the file to be uploaded

- **metadata** – This field is the metadata associated with the file. Description and tags are compulsory fields of metadata.
- **access_token** – This is user specific access token/master token.
- **base_url** – It is this URL *https://www.openhumans.org*.
- **remote_file_info** – This field is for checking if a file with matching name and file size already exists. Its default value is none.
- **project_member_id** – This field is the list of project member id of all members of a project. Its default value is None.
- **max_bytes** – This field is the maximum file size a user can upload. It's default value is 128m.

```
ohapi.api.upload_file(target_filepath, metadata, access_token, datatypes=None,
                      base_url='https://www.openhumans.org/', remote_file_info=None,
                      project_member_id=None, max_bytes=128000000)
```

Upload a file from a local filepath using the “direct upload” API. To learn more about this API endpoint see: * <https://www.openhumans.org/direct-sharing/on-site-data-upload/> * <https://www.openhumans.org/direct-sharing/oauth2-data-upload/>

Parameters

- **target_filepath** – This field is the filepath of the file to be uploaded
- **metadata** – This field is a python dictionary with keys filename, description and tags for single user upload and filename, project member id, description and tags for multiple user upload.
- **access_token** – This is user specific access token/master token.
- **base_url** – It is this URL *https://www.openhumans.org*.
- **remote_file_info** – This field is for for checking if a file with matching name and file size already exists. Its default value is none.
- **project_member_id** – This field is the list of project member id of all members of a project. Its default value is None.
- **max_bytes** – This field is the maximum file size a user can upload. It's default value is 128m.

```
ohapi.api.upload_stream(stream, filename, metadata, access_token, datatypes=None,
                        base_url='https://www.openhumans.org/', remote_file_info=None,
                        project_member_id=None, max_bytes=128000000, file_identifier=None)
```

Upload a file object using the “direct upload” feature, which uploads to an S3 bucket URL provided by the Open Humans API. To learn more about this API endpoint see: * <https://www.openhumans.org/direct-sharing/on-site-data-upload/> * <https://www.openhumans.org/direct-sharing/oauth2-data-upload/>

Parameters

- **stream** – This field is the stream (or file object) to be uploaded.
- **metadata** – This field is the metadata associated with the file. Description and tags are compulsory fields of metadata.
- **access_token** – This is user specific access token/master token.
- **base_url** – It is this URL *https://www.openhumans.org*.
- **remote_file_info** – This field is for for checking if a file with matching name and file size already exists. Its default value is none.

- **project_member_id** – This field is the list of project member id of all members of a project. Its default value is None.
- **max_bytes** – This field is the maximum file size a user can upload. Its default value is 128m.
- **max_bytes** – If provided, this is used in logging output. Its default value is None (in which case, filename is used).

1.2.2 ohapi.public module

Utility functions to download public data files.

`ohapi.public.download` (*source=None, username=None, directory='.', max_size='128m', quiet=None, debug=None*)

Download public data from Open Humans.

Parameters

- **source** – This field is the data source from which to download. It's default value is None.
- **username** – This field is username of user. It's default value is None.
- **directory** – This field is the target directory to which data is downloaded.
- **max_size** – This field is the maximum file size. It's default value is 128m.
- **quiet** – This field is the logging level. It's default value is None.
- **debug** – This field is the logging level. It's default value is None.

`ohapi.public.download_url` (*result, directory, max_bytes*)

Download a file.

Parameters

- **result** – This field contains a url from which data will be downloaded.
- **directory** – This field is the target directory to which data will be downloaded.
- **max_bytes** – This field is the maximum file size in bytes.

`ohapi.public.get_members_by_source` (*base_url='https://www.openhumans.org/api/public-data'*)

Function returns which members have joined each activity.

Parameters **base_url** – It is URL: *https://www.openhumans.org/api/public-data*.

`ohapi.public.get_sources_by_member` (*base_url='https://www.openhumans.org/api/public-data', limit=100*)

Function returns which activities each member has joined.

Parameters

- **base_url** – It is URL: *https://www.openhumans.org/api/public-data*.
- **limit** – It is the limit of data send by one request.

`ohapi.public.signal_handler_cb` (*signal_name, frame*)

Exit on Ctrl-C.

1.2.3 ohapi.projects module

Utility functions to use master_access_tokens to interact with a project

class ohapi.projects.OHProject (*master_access_token*)

Work with an Open Humans Project.

download_all (*target_dir, source=None, project_data=False, memberlist=None, excludelist=None, max_size='128m', id_filename=False*)

Download data for all users including shared data files.

Parameters

- **target_dir** – This field is the target directory to download data.
- **source** – This field is the data source. It's default value is None.
- **project_data** – This field is data related to particular project. It's default value is False.
- **memberlist** – This field is list of members whose data will be downloaded. It's default value is None.
- **excludelist** – This field is list of members whose data will be skipped. It's default value is None.
- **max_size** – This field is the maximum file size. It's default value is 128m.

classmethod download_member_project_data (*member_data, target_member_dir, max_size='128m', id_filename=False*)

Download files to sync a local dir to match OH member project data.

Parameters

- **member_data** – This field is data related to member in a project.
- **target_member_dir** – This field is the target directory where data will be downloaded.
- **max_size** – This field is the maximum file size. It's default value is 128m.

classmethod download_member_shared (*member_data, target_member_dir, source=None, max_size='128m', id_filename=False*)

Download files to sync a local dir to match OH member shared data.

Files are downloaded to match their “basename” on Open Humans. If there are multiple files with the same name, the most recent is downloaded.

Parameters

- **member_data** – This field is data related to member in a project.
- **target_member_dir** – This field is the target directory where data will be downloaded.
- **source** – This field is the source from which to download data.
- **max_size** – This field is the maximum file size. It's default value is 128m.

update_data ()

Returns data for all users including shared data files.

static upload_member_from_dir (*member_data, target_member_dir, metadata, access_token, mode='default', max_size='128m'*)

Upload files in target directory to an Open Humans member's account.

The default behavior is to overwrite files with matching filenames on Open Humans, but not otherwise delete files.

If the 'mode' parameter is 'safe': matching filenames will not be overwritten.

If the 'mode' parameter is 'sync': files on Open Humans that are not in the local directory will be deleted.

Parameters

- **member_data** – This field is data related to member in a project.
- **target_member_dir** – This field is the target directory from where data will be uploaded.
- **metadata** – This field is metadata for files to be uploaded.
- **access_token** – This field is user specific access token.
- **mode** – This field takes three value default, sync, safe. It's default value is 'default'.
- **max_size** – This field is the maximum file size. It's default value is 128m.

1.2.4 ohapi.utils_fs module

Utility functions to sync and work with Open Humans data in a local filesystem.

`ohapi.utils_fs.characterize_local_files` (*filedir, max_bytes=128000000*)

Collate local file info as preparation for Open Humans upload.

Note: Files with filesize > max_bytes are not included in returned info.

Parameters

- **filedir** – This field is target directory to get files from.
- **max_bytes** – This field is the maximum file size to consider. Its default value is 128m.

`ohapi.utils_fs.download_file` (*download_url, target_filepath, max_bytes=128000000*)

Download a file.

Parameters

- **download_url** – This field is the url from which data will be downloaded.
- **target_filepath** – This field is the path of the file where data will be downloaded.
- **max_bytes** – This field is the maximum file size to download. Its default value is 128m.

`ohapi.utils_fs.guess_tags` (*filename*)

Function to get potential tags for files using the file names.

Parameters filename – This field is the name of file.

`ohapi.utils_fs.is_single_file_metadata_valid` (*file_metadata, project_member_id, filename*)

Check if metadata fields like project member id, description, tags, md5 and creation date are valid for a single file.

Parameters

- **file_metadata** – This field is metadata of file.
- **project_member_id** – This field is the project member id corresponding to the file metadata provided.
- **filename** – This field is the filename corresponding to the file metadata provided.

`ohapi.utils_fs.load_metadata_csv(input_filepath)`

Return dict of metadata.

Format is either dict (filenames are keys) or dict-of-dicts (project member IDs as top level keys, then filenames as keys).

Parameters `input_filepath` – This field is the filepath of the csv file.

`ohapi.utils_fs.load_metadata_csv_multi_user(csv_in, header, tags_idx)`

Return the metadata as requested for multiple users.

Parameters

- `csv_in` – This field is the csv file to return metadata from.
- `header` – This field contains the headers in the csv file
- `tags_idx` – This field contains the index of the tags in the csv file.

`ohapi.utils_fs.load_metadata_csv_single_user(csv_in, header, tags_idx)`

Return the metadata as requested for a single user.

Parameters

- `csv_in` – This field is the csv file to return metadata from.
- `header` – This field contains the headers in the csv file
- `tags_idx` – This field contains the index of the tags in the csv file.

`ohapi.utils_fs.mk_metadata_csv(filedir, outputfilepath, max_bytes=128000000)`

Make metadata file for all files in a directory.

Parameters

- `filedir` – This field is the filepath of the directory whose csv has to be made.
- `outputfilepath` – This field is the file path of the output csv.
- `max_bytes` – This field is the maximum file size to consider. Its default value is 128m.

`ohapi.utils_fs.print_error(e)`

Helper function to print error.

Parameters `e` – This field is the error to be printed.

`ohapi.utils_fs.read_id_list(filepath)`

Get project member id from a file.

Parameters `filepath` – This field is the path of file to read.

`ohapi.utils_fs.review_metadata_csv(filedir, input_filepath)`

Check validity of metadata fields.

Parameters

- `filedir` – This field is the filepath of the directory whose csv has to be made.
- `outputfilepath` – This field is the file path of the output csv.
- `max_bytes` – This field is the maximum file size to consider. Its default value is 128m.

`ohapi.utils_fs.review_metadata_csv_multi_user(filedir, metadata, csv_in, n_headers)`

Check validity of metadata for multi user.

Parameters

- `filedir` – This field is the filepath of the directory whose csv has to be made.

- **metadata** – This field is the metadata generated from the `load_metadata_csv` function.
- **csv_in** – This field returns a reader object which iterates over the csv.
- **n_headers** – This field is the number of headers in the csv.

`ohapi.utils_fs.review_metadata_csv_single_user` (*filedir, metadata, csv_in, n_headers*)

Check validity of metadata for single user.

Parameters

- **filedir** – This field is the filepath of the directory whose csv has to be made.
- **metadata** – This field is the metadata generated from the `load_metadata_csv` function.
- **csv_in** – This field returns a reader object which iterates over the csv.
- **n_headers** – This field is the number of headers in the csv.

`ohapi.utils_fs.strip_zip_suffix` (*filename*)

Helper function to strip suffix from filename.

Parameters filename – This field is the name of file.

`ohapi.utils_fs.validate_date` (*date, project_member_id, filename*)

Check if date is in ISO 8601 format.

Parameters

- **date** – This field is the date to be checked.
- **project_member_id** – This field is the `project_member_id` corresponding to the date provided.
- **filename** – This field is the filename corresponding to the date provided.

`ohapi.utils_fs.validate_metadata` (*target_dir, metadata*)

Check that the files listed in metadata exactly match files in target dir.

Parameters

- **target_dir** – This field is the target directory from which to match metadata
- **metadata** – This field contains the metadata to be matched.

`ohapi.utils_fs.validate_subfolders` (*filedir, metadata*)

Check that all folders in the given directory have a corresponding entry in the metadata file, and vice versa.

Parameters

- **filedir** – This field is the target directory from which to match metadata
- **metadata** – This field contains the metadata to be matched.

`ohapi.utils_fs.write_metadata_to_filestream` (*filedir, filestream, max_bytes=128000000*)

Make metadata file for all files in a directory(helper function)

Parameters

- **filedir** – This field is the filepath of the directory whose csv has to be made.
- **filestream** – This field is a stream for writing to the csv.
- **max_bytes** – This field is the maximum file size to consider. Its default value is 128m.

1.3 Development & Testing

open-humans-api is under active development and we have implemented a couple of tests to check that changes don't break anything.

If you want to contribute you can do so on [GitHub](#) and find us on [Slack](#).

1.3.1 Installing ohapi for development

To install a development version of this package run the following steps on your shell:

```
git clone git@github.com:OpenHumans/open-humans-api.git
cd open-humans-api
pip install -e.
```

1.3.2 ohapi.tests.test_test

1.3.3 ohapi.tests.test_api

1.3.4 ohapi.tests.test_projects

CHAPTER 2

Indices and tables

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

O

`ohapi.api`, 12
`ohapi.command_line`, 9
`ohapi.projects`, 16
`ohapi.public`, 15
`ohapi.utils_fs`, 17

Symbols

- all_files <all_files>
 - ohproj-delete command line option,9
- all_members <all_members>
 - ohproj-message command line option,9
- base_url <base_url>
 - ohproj-oauth2-token-exchange command line option,8
 - ohproj-oauth2-url command line option,8
- code <code>
 - ohproj-oauth2-token-exchange command line option,8
- create-csv <create_csv>
 - ohproj-upload-metadata command line option,7
- debug
 - ohproj-download command line option,6
 - ohproj-download-metadata command line option,6
 - ohproj-message command line option,9
 - ohproj-upload command line option,7
 - ohproj-upload-metadata command line option,7
 - ohpub-download command line option,5
- excludelist <excludelist>
 - ohproj-download command line option,6
- id-filename
 - ohproj-download command line option,6
- max-size <max_size>
 - ohproj-download command line option,6
 - ohproj-upload command line option,7
- ohproj-upload-metadata command line option,7
- memberlist <memberlist>
 - ohproj-download command line option,6
- metadata-csv <metadata_csv>
 - ohproj-upload command line option,7
- output-csv <output_csv>
 - ohproj-download-metadata command line option,6
- project-data
 - ohproj-download command line option,6
- project_member_ids <project_member_ids>
 - ohproj-message command line option,9
- review <review>
 - ohproj-upload-metadata command line option,7
- safe
 - ohproj-upload command line option,7
- sync
 - ohproj-upload command line option,7
- T, -access_token <access_token>
 - ohproj-delete command line option,9
- T, -master-token <master_token>
 - ohproj-download command line option,5
 - ohproj-download-metadata command line option,6
 - ohproj-upload command line option,7
- at, -access_token <access_token>
 - ohproj-message command line option,9
- b, -file_basename <file_basename>
 - ohproj-delete command line option,9
- c, -client_id <client_id>
 - ohproj-oauth2-url command line option,8

-cid, -client_id <client_id>
 ohproj-oauth2-token-exchange
 command line option, 8

-cs, -client_secret <client_secret>
 ohproj-oauth2-token-exchange
 command line option, 8

-d, -directory <directory>
 ohproj-download command line
 option, 5
 ohproj-upload command line option, 7
 ohproj-upload-metadata command
 line option, 7
 ohpub-download command line option,
 5

-i, -file_id <file_id>
 ohproj-delete command line option, 9

-m, -max-size <max_size>
 ohpub-download command line option,
 5

-m, -member <member>
 ohproj-download command line
 option, 5
 ohproj-upload command line option, 7

-m, -message_body <message_body>
 ohproj-message command line option,
 9

-m, -project_member_id
 <project_member_id>
 ohproj-delete command line option, 9

-q, -quiet
 ohpub-download command line option,
 5

-r, -redirect_uri <redirect_uri>
 ohproj-oauth2-url command line
 option, 8

-re_uri, -redirect_uri <redirect_uri>
 ohproj-oauth2-token-exchange
 command line option, 8

-rt, -refresh_token <refresh_token>
 ohproj-oauth2-token-exchange
 command line option, 8

-s, -source <source>
 ohproj-download command line
 option, 6
 ohpub-download command line option,
 5

-s, -subject <subject>
 ohproj-message command line option,
 9

-t, -access-token <access_token>
 ohproj-download command line
 option, 6
 ohproj-upload command line option, 7

-u, -username <username>

ohpub-download command line option,
 5

-v, -verbose
 ohproj-download command line
 option, 6
 ohproj-download-metadata command
 line option, 6
 ohproj-message command line option,
 9
 ohproj-upload command line option, 7
 ohproj-upload-metadata command
 line option, 7

C

characterize_local_files() (in module
ohapi.utils_fs), 17

D

delete_file() (in module *ohapi.api*), 12
 delete_files() (in module *ohapi.api*), 12
 download() (in module *ohapi.command_line*), 9
 download() (in module *ohapi.public*), 15
 download_all() (*ohapi.projects.OHProject* method),
 16
 download_file() (in module *ohapi.utils_fs*), 17
 download_member_project_data()
 (*ohapi.projects.OHProject* class method),
 16
 download_member_shared()
 (*ohapi.projects.OHProject* class method),
 16
 download_metadata() (in module
ohapi.command_line), 10
 download_url() (in module *ohapi.public*), 15

E

exchange_oauth2_member() (in module
ohapi.api), 12

G

get_all_results() (in module *ohapi.api*), 12
 get_members_by_source() (in module
ohapi.public), 15
 get_page() (in module *ohapi.api*), 12
 get_sources_by_member() (in module
ohapi.public), 15
 guess_tags() (in module *ohapi.utils_fs*), 17

H

handle_error() (in module *ohapi.api*), 12

I

is_single_file_metadata_valid() (in mod-
ule ohapi.utils_fs), 17

L

load_metadata_csv() (in module *ohapi.utils_fs*), 17

load_metadata_csv_multi_user() (in module *ohapi.utils_fs*), 18

load_metadata_csv_single_user() (in module *ohapi.utils_fs*), 18

M

message() (in module *ohapi.api*), 12

mk_metadata_csv() (in module *ohapi.utils_fs*), 18

O

oauth2_auth_url() (in module *ohapi.api*), 13

oauth2_token_exchange() (in module *ohapi.api*), 13

ohapi.api (module), 12

ohapi.command_line (module), 9

ohapi.projects (module), 16

ohapi.public (module), 15

ohapi.utils_fs (module), 17

ohproj-delete command line option

- all_files <all_files>, 9
- T, -access_token <access_token>, 9
- b, -file_basename <file_basename>, 9
- i, -file_id <file_id>, 9
- m, -project_member_id <project_member_id>, 9

ohproj-download command line option

- debug, 6
- excludelist <excludelist>, 6
- id-filename, 6
- max-size <max_size>, 6
- memberlist <memberlist>, 6
- project-data, 6
- T, -master-token <master_token>, 5
- d, -directory <directory>, 5
- m, -member <member>, 5
- s, -source <source>, 6
- t, -access-token <access_token>, 6
- v, -verbose, 6

ohproj-download-metadata command line option

- debug, 6
- output-csv <output_csv>, 6
- T, -master-token <master_token>, 6
- v, -verbose, 6

ohproj-message command line option

- all_members <all_members>, 9
- debug, 9
- project_member_ids <project_member_ids>, 9

- at, -access_token <access_token>, 9

- m, -message_body <message_body>, 9

- s, -subject <subject>, 9

- v, -verbose, 9

ohproj-oauth2-token-exchange command line option

- base_url <base_url>, 8
- code <code>, 8
- cid, -client_id <client_id>, 8
- cs, -client_secret <client_secret>, 8
- re_uri, -redirect_uri <redirect_uri>, 8
- rt, -refresh_token <refresh_token>, 8

ohproj-oauth2-url command line option

- base_url <base_url>, 8
- c, -client_id <client_id>, 8
- r, -redirect_uri <redirect_uri>, 8

ohproj-upload command line option

- debug, 7
- max-size <max_size>, 7
- metadata-csv <metadata_csv>, 7
- safe, 7
- sync, 7
- T, -master-token <master_token>, 7
- d, -directory <directory>, 7
- m, -member <member>, 7
- t, -access-token <access_token>, 7
- v, -verbose, 7

ohproj-upload-metadata command line option

- create-csv <create_csv>, 7
- debug, 7
- max-size <max_size>, 7
- review <review>, 7
- d, -directory <directory>, 7
- v, -verbose, 7

OHPProject (class in *ohapi.projects*), 16

ohpub-download command line option

- debug, 5
- d, -directory <directory>, 5
- m, -max-size <max_size>, 5
- q, -quiet, 5
- s, -source <source>, 5
- u, -username <username>, 5

P

print_error() (in module *ohapi.utils_fs*), 18

R

read_id_list() (in module *ohapi.utils_fs*), 18

review_metadata_csv() (in module *ohapi.utils_fs*), 18

`review_metadata_csv_multi_user()` (in module `ohapi.utils_fs`), 18

`review_metadata_csv_single_user()` (in module `ohapi.utils_fs`), 19

S

`set_log_level()` (in module `ohapi.command_line`), 10

`SettingsError`, 12

`signal_handler_cb()` (in module `ohapi.public`), 15

`strip_zip_suffix()` (in module `ohapi.utils_fs`), 19

U

`update_data()` (`ohapi.projects.OHProject` method), 16

`upload()` (in module `ohapi.command_line`), 10

`upload_aws()` (in module `ohapi.api`), 13

`upload_file()` (in module `ohapi.api`), 14

`upload_member_from_dir()` (`ohapi.projects.OHProject` static method), 16

`upload_metadata()` (in module `ohapi.command_line`), 11

`upload_stream()` (in module `ohapi.api`), 14

V

`validate_date()` (in module `ohapi.utils_fs`), 19

`validate_metadata()` (in module `ohapi.utils_fs`), 19

`validate_subfolders()` (in module `ohapi.utils_fs`), 19

W

`write_metadata_to_filestream()` (in module `ohapi.utils_fs`), 19