Mendeley Python SDK Documentation

Release 0.3.1

Mendeley

March 11, 2015

Contents

1 Installation **3**

2 Compatibility **5**

3 Sample applications **7**

4 Contents **9**

4.1 Using the SDK 9

4.2 Resources 11

4.3 Models 18

The Mendeley Python SDK provides access to the Mendeley API. For more information on the API and its capabilities, see the developer portal.

CHAPTER 1

Installation

Install from PyPI using pip, a package manager for Python.

```
pip install mendeley
```

CHAPTER 2

Compatibility

The SDK has been tested with Python 2.7 and 3.4.

CHAPTER 3

Sample applications

There are two sample applications that illustrate how to use the SDK:

- mendeley-api-python-example, a Flask web application for accessing a user's library.
- mendeley-api-python-catalog-example, a command-line application for accessing the Mendeley catalog.

CHAPTER 4

Contents

4.1 Using the SDK

4.1.1 Authentication

There are three ways to authenticate with the Mendeley Python SDK. Before you start, you'll need to have registered your application at the developer portal.

Authorization code flow

This flow is recommended for applications that have access to secure, private storage, such as web applications deployed on a server.

```
from mendeley import Mendeley

# These values should match the ones supplied when registering your application.
mendeley = Mendeley(client_id, client_secret=client_secret, redirect_uri=redirect_uri)

auth = mendeley.start_authorization_code_flow()

# The user needs to visit this URL, and log in to Mendeley.
login_url = auth.get_login_url()

# After logging in, the user will be redirected to a URL, auth_response.
session = auth.authenticate(auth_response)
```

Implicit grant flow

This flow is recommended for applications running in environments that do not provide secure storage.

```
from mendeley import Mendeley

# These values should match the ones supplied when registering your application.
mendeley = Mendeley(client_id, redirect_uri=redirect_uri)

auth = mendeley.start_implicit_grant_flow()

# The user needs to visit this URL, and log in to Mendeley.
login_url = auth.get_login_url()
```

```
# After logging in, the user will be redirected to a URL, auth_response.
session = auth.authenticate(auth_response)
```

Client credentials flow

This flow does not require the user to log in. However, it only provides access to a limited set of resources (the read-only Mendeley Catalog of crowd sourced documents).

```
from mendeley import Mendeley

# These values should match the ones supplied when registering your application.
mendeley = Mendeley(client_id, client_secret=client_secret)

auth = mendeley.start_client_credentials_flow()
session = auth.authenticate()
```

4.1.2 Sessions

After authentication, you will have a `MendeleySession`, which will allow you to access the Mendeley API. The linked resources describe the operations that you can perform, and the objects that you can interact with.

class `mendeley.session.`**MendeleySession**

Entry point for accessing Mendeley resources.

annotations

A :class: *Annotations <mendeley.resources.annotations.Annotations>* resource for accessing annotations in the logged-in user's library.

catalog

A `Catalog` resource for accessing the Mendeley catalog.

documents

A `Documents` resource for accessing documents in the logged-in user's library.

files

A `Files` resource for accessing files in the logged-in user's library.

groups

A `Groups` resource for accessing groups that the user is a member of.

profiles

A `Profiles` resource for accessing profiles of Mendeley users.

trash

A `Trash` resource for accessing trashed documents in the logged-in user's library.

4.1.3 Pagination

Many collections in the API are spread over multiple pages. Typically, the SDK provides two ways of navigating these collections:

- an *iter* method, which provides the whole collection as an iterator.
- a *list* method, which returns the first page of results as a `Page`. You can use this object to navigate through the other pages.

class mendeley.pagination.**Page**
A page of a collection of objects.

first_page
the first Page of the collection.

items
a list of items on this page.

last_page
the first Page of the collection.

next_page
the next Page of the collection, or *None* if this is the last page.

previous_page
the previous Page of the collection, or *None* if this is the first page.

4.1.4 Examples

To print the name of the logged-in user:

```
print session.profiles.me.display_name
```

To print the titles of all of the documents in the user's library:

```
for document in session.documents.iter():
    print document.title
```

To print the number of readers of a document by DOI:

```
print session.catalog.by_identifier(doi='10.1371/journal.pmed.0020124', view='stats').reader_count
```

4.2 Resources

This page details the top-level resources that are available from a Mendeley session.

4.2.1 Documents

class mendeley.resources.documents.**Documents**
Top-level resource for accessing documents. These can be:

- documents for the logged-in user, if retrieved from a MendeleySession.
- documents in a Group.

advanced_search(*title=None*, *author=None*, *source=None*, *abstract=None*, *min_year=None*, *max_year=None*, *view=None*)
Executes an advanced search in the logged-in user's library, where individual fields can be searched on.

Parameters

- **title** – Title.
- **author** – Author.
- **source** – Source.
- **abstract** – Abstract.

- **min_year** – Minimum year for documents to return.
- **max_year** – Maximum year for documents to return.
- **view** – the view to get. One of 'bib', 'client', 'tags', 'all'.

Returns a `DocumentsSearch` resource, from which results can be retrieved.

create(*title*, *type*, ***kwargs*)
Creates a new document from metadata.

Parameters

- **title** – title of the document.
- **type** – type of the document.
- **kwargs** – other properties of the document. These can be any of the attributes on `UserDocument` or any of its views.

Returns a `UserDocument`.

create_from_file(*path*)
Creates a new document from a file.

Parameters **path** – path to the file.

Returns a `UserDocument`.

get(*id*, *view=None*)
Retrieves a document by ID.

Parameters

- **id** – the ID of the document to get.
- **view** – the view to get. One of 'bib', 'client', 'tags', 'all'.

Returns a `UserDocument`.

iter(*page_size=None*, *view=None*, *sort=None*, *order=None*, *modified_since=None*, *deleted_since=None*)
Retrieves documents, as an iterator.

Parameters

- **page_size** – the number of documents to retrieve at a time. Defaults to 20.
- **view** – the view to get. One of 'bib', 'client', 'tags', 'all'.
- **sort** – if specified, sorts documents by the specified field. One of 'created', 'last_modified', 'title'.
- **order** – if specified in conjunction with 'sort', specifies the sort order. One of 'asc', 'desc'.
- **modified_since** – if specified, only returns files modified after this timestamp.
- **deleted_since** – if specified, only returns the IDs of documents deleted after this timestamp.

Returns an iterator of `UserDocuments`.

list(*page_size=None*, *view=None*, *sort=None*, *order=None*, *modified_since=None*, *deleted_since=None*)
Retrieves documents, as a paginated collection.

Parameters

- **page_size** – the number of documents to return on each page. Defaults to 20.

- **view** – the view to get. One of 'bib', 'client', 'tags', 'all'.
- **sort** – if specified, sorts documents by the specified field. One of 'created', 'last_modified', 'title'.
- **order** – if specified in conjunction with 'sort', specifies the sort order. One of 'asc', 'desc'.
- **modified_since** – if specified, only returns files modified after this timestamp.
- **deleted_since** – if specified, only returns the IDs of documents deleted after this timestamp.

Returns a `Page` of `UserDocuments`.

search(*query*, *view=None*)
Searches the logged-in user's library for documents.

Parameters

- **query** – the search query to execute.
- **view** – the view to get. One of 'bib', 'client', 'tags', 'all'.

Returns a `DocumentsSearch` resource, from which results can be retrieved.

class `mendeley.resources.trash.`**Trash**
Top-level resource for accessing trashed documents. These can be:

•trashed documents for the logged-in user, if retrieved from a `MendeleySession`.

•trashed documents in a `Group`.

get(*id*, *view=None*)
Retrieves a trashed document by ID.

Parameters

- **id** – the ID of the document to get.
- **view** – the view to get. One of 'bib', 'client', 'tags', 'all'.

Returns a `TrashDocument`.

iter(*page_size=None*, *view=None*, *sort=None*, *order=None*, *modified_since=None*, *deleted_since=None*)
Retrieves trashed documents, as an iterator.

Parameters

- **page_size** – the number of documents to retrieve at a time. Defaults to 20.
- **view** – the view to get. One of 'bib', 'client', 'tags', 'all'.
- **sort** – if specified, sorts documents by the specified field. One of 'created', 'last_modified', 'title'.
- **order** – if specified in conjunction with 'sort', specifies the sort order. One of 'asc', 'desc'.
- **modified_since** – if specified, only returns files modified after this timestamp.
- **deleted_since** – if specified, only returns the IDs of documents deleted after this timestamp.

Returns an iterator of `TrashDocuments`.

list(*page_size=None*, *view=None*, *sort=None*, *order=None*, *modified_since=None*, *deleted_since=None*)
Retrieves trashed documents, as a paginated collection.

Parameters

- **page_size** – the number of documents to return on each page. Defaults to 20.
- **view** – the view to get. One of 'bib', 'client', 'tags', 'all'.
- **sort** – if specified, sorts documents by the specified field. One of 'created', 'last_modified', 'title'.
- **order** – if specified in conjunction with 'sort', specifies the sort order. One of 'asc', 'desc'.
- **modified_since** – if specified, only returns files modified after this timestamp.
- **deleted_since** – if specified, only returns the IDs of documents deleted after this timestamp.

Returns a `Page` of `TrashDocuments`.

4.2.2 Catalog

class `mendeley.resources.catalog.`**`Catalog`**

Top-level resource for accessing catalog documents.

`advanced_search`(*title=None*, *author=None*, *source=None*, *abstract=None*, *min_year=None*, *max_year=None*, *open_access=None*, *view=None*)

Executes an advanced catalog search, where individual fields can be searched on.

Parameters

- **title** – Title.
- **author** – Author.
- **source** – Source.
- **abstract** – Abstract.
- **min_year** – Minimum year for documents to return.
- **max_year** – Maximum year for documents to return.
- **open_access** – If 'true', only returns open access documents.
- **view** – the view to get. One of 'bib', 'client', 'stats', 'all'.

Returns a `CatalogSearch` resource, from which results can be retrieved.

`by_identifier`(*arxiv=None*, *doi=None*, *isbn=None*, *issn=None*, *pmid=None*, *scopus=None*, *filehash=None*, *view=None*)

Retrieves a catalog document by an external identifier. Only one identifier may be specified.

Parameters

- **arxiv** – ArXiV ID.
- **doi** – DOI.
- **isbn** – ISBN.
- **issn** – ISSN.
- **pmid** – PubMed ID.
- **scopus** – Scopus ID (EID).
- **filehash** – SHA-1 filehash.

- **view** – the view to get. One of 'bib', 'client', 'stats', 'all'.

Returns a `CatalogDocument`.

get(*id*, *view=None*)
Retrieves a catalog document by ID.

Parameters

- **id** – the ID of the document to get.
- **view** – the view to get. One of 'bib', 'client', 'stats', 'all'.

Returns a `CatalogDocument`.

lookup(*arxiv=None*, *doi=None*, *pmid=None*, *filehash=None*, *title=None*, *authors=None*, *year=None*, *source=None*, *view=None*)
Finds the closest matching catalog document to a supplied set of metadata.

Parameters

- **arxiv** – ArXiV ID.
- **doi** – DOI.
- **pmid** – PubMed ID.
- **filehash** – SHA-1 filehash.
- **title** – Title.
- **authors** – Authors.
- **year** – Year.
- **source** – Source.
- **view** – the view to get. One of 'bib', 'client', 'stats', 'all'.

Returns a `CatalogDocument`.

search(*query*, *view=None*)
Searches the catalog for documents.

Parameters

- **query** – the search query to execute.
- **view** – the view to get. One of 'bib', 'client', 'stats', 'all'.

Returns a `CatalogSearch` resource, from which results can be retrieved.

class mendeley.resources.catalog.**CatalogSearch**
Resource for accessing the results of a catalog search.

iter(*page_size=None*)
Retrieves search results, as an iterator.

Parameters **page_size** – the number of search results to retrieve at a time. Defaults to 20.

Returns an iterator of `CatalogDocuments`.

list(*page_size=None*)
Retrieves search results, as a paginated collection.

Parameters **page_size** – the number of search results to return on each page. Defaults to 20.

Returns a `Page` of `CatalogDocuments`.

4.2.3 Files

class mendeley.resources.files.**Files**
Top-level resource for accessing files. These can be:

- files for the logged-in user, if retrieved from a MendeleySession.
- files attached to a CatalogDocument.
- files attached to a UserDocument.
- files in a Group.

iter(*page_size=None*, *added_since=None*, *deleted_since=None*)
Retrieves files, as an iterator.

Parameters

- **page_size** – the number of files to retrieve at a time. Defaults to 20.
- **added_since** – if specified, only returns files added after this timestamp.
- **deleted_since** – if specified, only returns the IDs of files deleted after this timestamp.

Returns an iterator of Files.

list(*page_size=None*, *added_since=None*, *deleted_since=None*)
Retrieves files, as a paginated collection.

Parameters

- **page_size** – the number of files to return on each page. Defaults to 20.
- **added_since** – if specified, only returns files added after this timestamp.
- **deleted_since** – if specified, only returns the IDs of files deleted after this timestamp.

Returns a Page of Files.

4.2.4 Groups

class mendeley.resources.groups.**Groups**
Top-level resource for accessing groups.

get(*id*)
Retrieves a group by ID.

Parameters **id** – the ID of the group to get.

Returns a Group.

iter(*page_size=None*)
Retrieves groups that the logged-in user is a member of, as an iterator.

Parameters **page_size** – the number of groups to retrieve at a time. Defaults to 20.

Returns an iterator of Groups.

list(*page_size=None*)
Retrieves groups that the logged-in user is a member of, as a paginated collection.

Parameters **page_size** – the number of groups to return on each page. Defaults to 20.

Returns a Page of Groups.

class mendeley.resources.groups.**GroupMembers**
Resource for accessing members of a group.

iter(*page_size=None*)
Retrieves members of the group, as an iterator.

Parameters **page_size** – the number of members to retrieve at a time. Defaults to 20.

Returns an iterator of GroupMembers.

list(*page_size=None*)
Retrieves members of the group, as a paginated collection.

Parameters **page_size** – the number of members to return on each page. Defaults to 20.

Returns a Page of GroupMembers.

4.2.5 Profiles

class mendeley.resources.profiles.**Profiles**
Top-level resource for accessing profiles.

get(*id*)
Retrieves a profile by ID.

Parameters **id** – the ID of the profile to get.

Returns a Profile.

me
The Profile of the logged-in user.

4.2.6 Annotations

class mendeley.resources.annotations.**Annotations**
Top-level resource for accessing annotations.

get(*id*)
Retrieves an annotation by ID.

Parameters **id** – the ID of the annotation to get.

Returns a Annotation.

iter(*page_size=None*, *modified_since=None*, *deleted_since=None*)
Retrieves annotations that the logged-in user is a member of, as an iterator.

Parameters

- **page_size** – the number of annotations to retrieve at a time. Defaults to 20.
- **modified_since** – if specified, only returns annotations modified after this timestamp.
- **deleted_since** – if specified, only returns the annotations deleted after this timestamp.

Returns an iterator of Annotations.

list(*page_size=None*, *modified_since=None*, *deleted_since=None*)
Retrieves annotations as a paginated collection.

Parameters

- **page_size** – the number of annotations to return on each page. Defaults to 20.

- **modified_since** – if specified, only returns annotations modified after this timestamp.
- **deleted_since** – if specified, only returns annotations deleted after this timestamp.

Returns a `Page` of `Annotations`.

4.3 Models

This page details the various model objects in the Mendeley Python SDK.

4.3.1 User documents

Documents

class `mendeley.models.documents.`**UserDocument**

Base class for user documents.

id

title

type

source

year

identifiers

keywords

abstract

add_note(*text*)

Adds a note to this document.

Parameters **text** – the text of the note to add.

Returns a `Annotation`.

attach_file(*path*)

Attaches a file to this document.

Parameters **path** – the path of the file to attach.

Returns a `File`.

authors

a list of `Persons`.

created

an `Arrow` object.

delete()

Permanently deletes this document.

files

a `Files` resource, from which `Files` can be retrieved.

group
a Group.

last_modified
an Arrow object.

move_to_trash()
Moves this document to the trash.

Returns a TrashDocument.

profile
a Profile.

update(***kwargs*)
Updates this document.

Parameters **kwargs** – updated field values. Only the values supplied will be modified.

Returns the updated document.

class mendeley.models.documents.**TrashDocument**
Base class for trashed documents.

id

title

type

source

year

identifiers

keywords

abstract

authors
a list of Persons.

created
an Arrow object.

delete()
Permanently deletes this document.

files
a Files resource, from which Files can be retrieved.

group
a Group.

last_modified
an Arrow object.

profile
a Profile.

restore()
Restores this document from the trash.

Returns a UserDocument.

Views

The following views are available on `UserDocuments` and `TrashDocuments`.

class `mendeley.models.documents.`**`UserBibView`**
Additional fields returned when getting a `UserDocument` or `TrashDocument` with view='bib' or 'all'.

`pages`

`volume`

`issue`

`websites`

`month`

`publisher`

`day`

`city`

`edition`

`institution`

`series`

`chapter`

`revision`

`accessed`
an `Arrow` object.

`editors`
a list of `Persons`.

class `mendeley.models.documents.`**`UserClientView`**
Additional fields returned when getting a `UserDocument` or `TrashDocument` with view='client' or 'all'.

`file_attached`

`read`

`starred`

`authored`

`confirmed`

`hidden`

class `mendeley.models.documents.`**`UserTagsView`**
Additional fields returned when getting a `UserDocument` or `TrashDocument` with view='tags' or 'all'.

`tags`

4.3.2 Catalog documents

Documents

class `mendeley.models.catalog.`**`CatalogDocument`**
Base class for catalog documents.

id

title

type

source

year

identifiers

keywords

abstract

link

authors
a list of `Persons`.

files
a `Files` resource, from which `Files` can be retrieved.

Views

The following views are available on `CatalogDocuments`.

class `mendeley.models.catalog.`**CatalogBibView**
Additional fields returned when getting a `CatalogDocument` with view='bib' or 'all'.

pages

volume

issue

websites

month

publisher

day

city

edition

institution

series

chapter

revision

editors
a list of `Persons`.

class `mendeley.models.catalog.`**CatalogClientView**
Additional fields returned when getting a `CatalogDocument` with view='client' or 'all'.

file_attached

class `mendeley.models.catalog.`**CatalogStatsView**
Additional fields returned when getting a `CatalogDocument` with view='stats' or 'all'.

reader_count

reader_count_by_academic_status

reader_count_by_subdiscipline

reader_count_by_country

4.3.3 Profiles

class mendeley.models.profiles.**Profile**
A Mendeley profile.

id

first_name

last_name

display_name

email

link

research_interests

academic_status

verified

user_type

created
an Arrow object.

discipline
a Discipline.

education
a list of Education objects.

employment
a list of Employment objects.

location
a Location.

photo
a Photo.

4.3.4 Groups

class mendeley.models.groups.**Group**
A Mendeley group.

id

name

description

disciplines

tags

webpage

link

access_level

role

created
an Arrow object.

documents
a Documents resource, from which UserDocuments can be retrieved.

files
a Files resource, from which Files can be retrieved.

owner
a Profile.

photo
a Photo.

trash
a Trash resource, from which TrashDocuments can be retrieved.

class mendeley.models.groups.**GroupMember**
A member of a Mendeley group.

id

role

first_name

last_name

display_name

email

link

research_interests

academic_status

verified

user_type

created
an Arrow object.

discipline
a Discipline.

education
a list of Education objects.

employment
a list of Employment objects.

joined
an Arrow object.

location
a Location.

photo
a Photo.

4.3.5 Files

class mendeley.models.files.**File**
A file attached to a document.

id

size

file_name

mime_type

filehash

download_url

add_highlight(*bounding_boxes*, *color*)
Adds a highlight to this file.

Parameters

- **bounding_boxes** – the area the highlight covers on the file.
- **color** – the color of the highlight.

Returns a Annotation.

add_sticky_note(*text*, *x_position*, *y_position*, *page_number*)
Adds a sticky note to this file.

Parameters

- **text** – the text of the sticky_note.
- **x_position** – the x position on the file of the sticky_note.
- **y_position** – the y position on the file of the stick_note.
- **page_number** – the page_number on the file of the sticky_note.

Returns a Annotation.

delete()
Deletes the file.

document(*view=None*)

Parameters **view** – document view to return.

Returns a UserDocument or CatalogDocument, depending on which the document is attached to.

download(*directory*)
Downloads the file.

Parameters **directory** – the directory to download the file to. This must exist.

Returns the path to the downloaded file.

download_url
the URL at which the file can be downloaded. This is only valid for a short time, so should not be cached.

4.3.6 Annotations

class mendeley.models.annotations.**Annotation**
An annotation attached to a document or a file.

id

type

text

privacy_level

color
a Color.

created
an Arrow object.

delete()
Permanently deletes this annotation.

document(*view=None*)

Parameters **view** – document view to return.

Returns a UserDocument

last_modified
an Arrow object.

positions
a list of BoundingBox.

profile
a Profile.

update(***kwargs*)
Updates this annotation.

4.3.7 Common objects

class mendeley.models.common.**Discipline**
The discipline of a Profile.

name

subdisciplines

class mendeley.models.common.**Education**
Education details, associated with a Profile.

institution

degree

website

end_date
an Arrow object.

start_date
an Arrow object.

class mendeley.models.common.**Employment**
Employment details, associated with a Profile.

institution

position

website

classes

end_date
an Arrow object.

start_date
an Arrow object.

class mendeley.models.common.**Location**
A location, associated with a Profile.

latitude

longitude

name

class mendeley.models.common.**Person**
A person, associated with a Document, as either an author or an editor.

first_name

last_name

static create(*first_name*, *last_name*)
Creates a person object, to be used when creating or updating a Document.

Parameters

- **first_name** –
- **last_name** –

Returns a Person.

class mendeley.models.common.**Photo**
A photo, associated with a Profile or Group.

original

standard

square

A

abstract (CatalogDocument attribute), 21
abstract (TrashDocument attribute), 19
abstract (UserDocument attribute), 18
academic_status (GroupMember attribute), 23
academic_status (Profile attribute), 22
access_level (Group attribute), 23
accessed (mendeley.models.documents.UserBibView attribute), 20
add_highlight() (mendeley.models.files.File method), 24
add_note() (mendeley.models.documents.UserDocument method), 18
add_sticky_note() (mendeley.models.files.File method), 24
advanced_search() (mendeley.resources.catalog.Catalog method), 14
advanced_search() (mendeley.resources.documents.Documents method), 11
Annotation (class in mendeley.models.annotations), 25
Annotations (class in mendeley.resources.annotations), 17
annotations (MendeleySession attribute), 10
attach_file() (mendeley.models.documents.UserDocument method), 18
authored (UserClientView attribute), 20
authors (mendeley.models.catalog.CatalogDocument attribute), 21
authors (mendeley.models.documents.TrashDocument attribute), 19
authors (mendeley.models.documents.UserDocument attribute), 18

B

by_identifier() (mendeley.resources.catalog.Catalog method), 14

C

Catalog (class in mendeley.resources.catalog), 14
catalog (MendeleySession attribute), 10
CatalogBibView (class in mendeley.models.catalog), 21
CatalogClientView (class in mendeley.models.catalog), 21
CatalogDocument (class in mendeley.models.catalog), 20
CatalogSearch (class in mendeley.resources.catalog), 15
CatalogStatsView (class in mendeley.models.catalog), 21
chapter (CatalogBibView attribute), 21
chapter (UserBibView attribute), 20
city (CatalogBibView attribute), 21
city (UserBibView attribute), 20
classes (Employment attribute), 26
color (mendeley.models.annotations.Annotation attribute), 25
confirmed (UserClientView attribute), 20
create() (mendeley.models.common.Person static method), 26
create() (mendeley.resources.documents.Documents method), 12
create_from_file() (mendeley.resources.documents.Documents method), 12
created (mendeley.models.annotations.Annotation attribute), 25
created (mendeley.models.documents.TrashDocument attribute), 19
created (mendeley.models.documents.UserDocument attribute), 18
created (mendeley.models.groups.Group attribute), 23
created (mendeley.models.groups.GroupMember attribute), 23
created (mendeley.models.profiles.Profile attribute), 22

D

day (CatalogBibView attribute), 21
day (UserBibView attribute), 20
degree (Education attribute), 25
delete() (mendeley.models.annotations.Annotation method), 25
delete() (mendeley.models.documents.TrashDocument method), 19

delete() (mendeley.models.documents.UserDocument method), 18
delete() (mendeley.models.files.File method), 24
description (Group attribute), 22
Discipline (class in mendeley.models.common), 25
discipline (mendeley.models.groups.GroupMember attribute), 23
discipline (mendeley.models.profiles.Profile attribute), 22
disciplines (Group attribute), 22
display_name (GroupMember attribute), 23
display_name (Profile attribute), 22
document() (mendeley.models.annotations.Annotation method), 25
document() (mendeley.models.files.File method), 24
Documents (class in mendeley.resources.documents), 11
documents (mendeley.models.groups.Group attribute), 23
documents (MendeleySession attribute), 10
download() (mendeley.models.files.File method), 24
download_url (File attribute), 24
download_url (mendeley.models.files.File attribute), 24

E

edition (CatalogBibView attribute), 21
edition (UserBibView attribute), 20
editors (mendeley.models.catalog.CatalogBibView attribute), 21
editors (mendeley.models.documents.UserBibView attribute), 20
Education (class in mendeley.models.common), 25
education (mendeley.models.groups.GroupMember attribute), 23
education (mendeley.models.profiles.Profile attribute), 22
email (GroupMember attribute), 23
email (Profile attribute), 22
Employment (class in mendeley.models.common), 26
employment (mendeley.models.groups.GroupMember attribute), 23
employment (mendeley.models.profiles.Profile attribute), 22
end_date (mendeley.models.common.Education attribute), 25
end_date (mendeley.models.common.Employment attribute), 26

F

File (class in mendeley.models.files), 24
file_attached (CatalogClientView attribute), 21
file_attached (UserClientView attribute), 20
file_name (File attribute), 24
filehash (File attribute), 24
Files (class in mendeley.resources.files), 16
files (mendeley.models.catalog.CatalogDocument attribute), 21
files (mendeley.models.documents.TrashDocument attribute), 19
files (mendeley.models.documents.UserDocument attribute), 18
files (mendeley.models.groups.Group attribute), 23
files (MendeleySession attribute), 10
first_name (GroupMember attribute), 23
first_name (Person attribute), 26
first_name (Profile attribute), 22
first_page (mendeley.pagination.Page attribute), 11

G

get() (mendeley.resources.annotations.Annotations method), 17
get() (mendeley.resources.catalog.Catalog method), 15
get() (mendeley.resources.documents.Documents method), 12
get() (mendeley.resources.groups.Groups method), 16
get() (mendeley.resources.profiles.Profiles method), 17
get() (mendeley.resources.trash.Trash method), 13
Group (class in mendeley.models.groups), 22
group (mendeley.models.documents.TrashDocument attribute), 19
group (mendeley.models.documents.UserDocument attribute), 18
GroupMember (class in mendeley.models.groups), 23
GroupMembers (class in mendeley.resources.groups), 16
Groups (class in mendeley.resources.groups), 16
groups (MendeleySession attribute), 10

H

hidden (UserClientView attribute), 20

I

id (Annotation attribute), 25
id (CatalogDocument attribute), 20
id (File attribute), 24
id (Group attribute), 22
id (GroupMember attribute), 23
id (Profile attribute), 22
id (TrashDocument attribute), 19
id (UserDocument attribute), 18
identifiers (CatalogDocument attribute), 21
identifiers (TrashDocument attribute), 19
identifiers (UserDocument attribute), 18
institution (CatalogBibView attribute), 21
institution (Education attribute), 25
institution (Employment attribute), 26
institution (UserBibView attribute), 20
issue (CatalogBibView attribute), 21
issue (UserBibView attribute), 20
items (mendeley.pagination.Page attribute), 11
iter() (mendeley.resources.annotations.Annotations method), 17

iter() (mendeley.resources.catalog.CatalogSearch method), 15
iter() (mendeley.resources.documents.Documents method), 12
iter() (mendeley.resources.files.Files method), 16
iter() (mendeley.resources.groups.GroupMembers method), 17
iter() (mendeley.resources.groups.Groups method), 16
iter() (mendeley.resources.trash.Trash method), 13

J

joined (mendeley.models.groups.GroupMember attribute), 23

K

keywords (CatalogDocument attribute), 21
keywords (TrashDocument attribute), 19
keywords (UserDocument attribute), 18

L

last_modified (mendeley.models.annotations.Annotation attribute), 25
last_modified (mendeley.models.documents.TrashDocument attribute), 19
last_modified (mendeley.models.documents.UserDocument attribute), 19
last_name (GroupMember attribute), 23
last_name (Person attribute), 26
last_name (Profile attribute), 22
last_page (mendeley.pagination.Page attribute), 11
latitude (Location attribute), 26
link (CatalogDocument attribute), 21
link (Group attribute), 23
link (GroupMember attribute), 23
link (Profile attribute), 22
list() (mendeley.resources.annotations.Annotations method), 17
list() (mendeley.resources.catalog.CatalogSearch method), 15
list() (mendeley.resources.documents.Documents method), 12
list() (mendeley.resources.files.Files method), 16
list() (mendeley.resources.groups.GroupMembers method), 17
list() (mendeley.resources.groups.Groups method), 16
list() (mendeley.resources.trash.Trash method), 13
Location (class in mendeley.models.common), 26
location (mendeley.models.groups.GroupMember attribute), 23
location (mendeley.models.profiles.Profile attribute), 22
longitude (Location attribute), 26
lookup() (mendeley.resources.catalog.Catalog method), 15

M

me (mendeley.resources.profiles.Profiles attribute), 17
MendeleySession (class in mendeley.session), 10
mime_type (File attribute), 24
month (CatalogBibView attribute), 21
month (UserBibView attribute), 20
move_to_trash() (mendeley.models.documents.UserDocument method), 19

N

name (Discipline attribute), 25
name (Group attribute), 22
name (Location attribute), 26
next_page (mendeley.pagination.Page attribute), 11

O

original (Photo attribute), 26
owner (mendeley.models.groups.Group attribute), 23

P

Page (class in mendeley.pagination), 10
pages (CatalogBibView attribute), 21
pages (UserBibView attribute), 20
Person (class in mendeley.models.common), 26
Photo (class in mendeley.models.common), 26
photo (mendeley.models.groups.Group attribute), 23
photo (mendeley.models.groups.GroupMember attribute), 24
photo (mendeley.models.profiles.Profile attribute), 22
position (Employment attribute), 26
positions (mendeley.models.annotations.Annotation attribute), 25
previous_page (mendeley.pagination.Page attribute), 11
privacy_level (Annotation attribute), 25
Profile (class in mendeley.models.profiles), 22
profile (mendeley.models.annotations.Annotation attribute), 25
profile (mendeley.models.documents.TrashDocument attribute), 19
profile (mendeley.models.documents.UserDocument attribute), 19
Profiles (class in mendeley.resources.profiles), 17
profiles (MendeleySession attribute), 10
publisher (CatalogBibView attribute), 21
publisher (UserBibView attribute), 20

R

read (UserClientView attribute), 20
reader_count (CatalogStatsView attribute), 21
reader_count_by_academic_status (CatalogStatsView attribute), 22

reader_count_by_country (CatalogStatsView attribute), 22
reader_count_by_subdiscipline (CatalogStatsView attribute), 22
research_interests (GroupMember attribute), 23
research_interests (Profile attribute), 22
restore() (mendeley.models.documents.TrashDocument method), 19
revision (CatalogBibView attribute), 21
revision (UserBibView attribute), 20
role (Group attribute), 23
role (GroupMember attribute), 23

S

search() (mendeley.resources.catalog.Catalog method), 15
search() (mendeley.resources.documents.Documents method), 13
series (CatalogBibView attribute), 21
series (UserBibView attribute), 20
size (File attribute), 24
source (CatalogDocument attribute), 21
source (TrashDocument attribute), 19
source (UserDocument attribute), 18
square (Photo attribute), 26
standard (Photo attribute), 26
starred (UserClientView attribute), 20
start_date (mendeley.models.common.Education attribute), 26
start_date (mendeley.models.common.Employment attribute), 26
subdisciplines (Discipline attribute), 25

T

tags (Group attribute), 22
tags (UserTagsView attribute), 20
text (Annotation attribute), 25
title (CatalogDocument attribute), 21
title (TrashDocument attribute), 19
title (UserDocument attribute), 18
Trash (class in mendeley.resources.trash), 13
trash (mendeley.models.groups.Group attribute), 23
trash (MendeleySession attribute), 10
TrashDocument (class in mendeley.models.documents), 19
type (Annotation attribute), 25
type (CatalogDocument attribute), 21
type (TrashDocument attribute), 19
type (UserDocument attribute), 18

U

update() (mendeley.models.annotations.Annotation method), 25
update() (mendeley.models.documents.UserDocument method), 19
user_type (GroupMember attribute), 23
user_type (Profile attribute), 22
UserBibView (class in mendeley.models.documents), 20
UserClientView (class in mendeley.models.documents), 20
UserDocument (class in mendeley.models.documents), 18
UserTagsView (class in mendeley.models.documents), 20

V

verified (GroupMember attribute), 23
verified (Profile attribute), 22
volume (CatalogBibView attribute), 21
volume (UserBibView attribute), 20

W

webpage (Group attribute), 23
website (Education attribute), 25
website (Employment attribute), 26
websites (CatalogBibView attribute), 21
websites (UserBibView attribute), 20

Y

year (CatalogDocument attribute), 21
year (TrashDocument attribute), 19
year (UserDocument attribute), 18