
Tokage Documentation

Release 1.1.0

Synder

Oct 02, 2018

Contents

1	Contents	3
1.1	Getting Started	3
1.1.1	Installing	3
1.1.2	Usage	3
1.1.3	Examples	3
1.2	Class Reference	5
1.2.1	Client	5
1.2.2	Base Classes	6
1.2.3	Partial Classes	10
1.3	Exceptions	11
1.3.1	Generic	11
1.3.2	Specific	11
2	Indices and tables	13

Tokage is an asyncio-based wrapper for the MAL api (using jikan.moe). This module features an easy to understand object-oriented style.

The source code can be found [here](#).

1.1 Getting Started

1.1.1 Installing

You can install Tokage using pip:

```
pip install -U tokage
```

1.1.2 Usage

Tokage is an *asynchronous* library. This means you must use some kind of async library to handle the HTTP requests to the API.

Tokage supports `asyncio`, which is provided in the standard python library.

Additionally, Tokage also supports `curio` and `trio`, two awesome libraries which solve many of the problems `asyncio` has. `multio`, a library that brings the two together, is required in order to use them.

Whichever you prefer, tokage will adapt to it. For more info on how to use each type of library, refer to the examples below.

1.1.3 Examples

Basic Asyncio Example

```
import asyncio
import tokage

async def main():
    client = tokage.Client() # Create a new Client instance. Default lib is asyncio.
```

(continues on next page)

(continued from previous page)

```
anime_id = await client.search_id("anime", "Steins Gate") # Search for an ID of
↳an Anime called "Steins gate"
anime = await client.get_anime(anime_id) # Get the Anime object from the API

manga_id = await client.search_id("manga", "my hero academia") # Search for an
↳ID of a Manga called "my hero academia"
manga = await client.get_manga(manga_id) # Get the Manga object from the API

print("Anime title:", anime.title) # Print the title of the Anime
print("Manga title:", manga.title) # Print the title of the Manga

await client.cleanup() # Finally, clean up

loop = asyncio.get_event_loop() # Get the event loop
loop.run_until_complete(main()) # Run main
```

Prints:

```
Anime title: Steins;Gate
Manga title: Boku no Hero Academia
```

Basic Multio Example

```
import multio
import tokage

multio.init('trio') # initialize multio to use trio. Also works the same with curio

async def main():
    client = tokage.Client(lib='multio') # Create a client instance. Tell tokage to
    ↳use multio (in this case trio)

    anime = await client.get_anime(1) # Get the Anime with ID `1`, which is Cowboy
    ↳Bebop

    print("Anime title:", anime.title) # Print the title and premiere season of the
    ↳Anime
    print("Started airing:", anime.premiered)

    await client.cleanup() # Finally, clean up (currently, this does nothing in
    ↳curio/trio mode, so it can be omitted)

multio.run(main) # Run main
```

Prints:

```
Anime title: Cowboy Bebop
Started airing: Spring 1998
```


1.2 Class Reference

1.2.1 Client

class tokage.**Client** (*session=None*, *, *lib='asyncio'*, *loop=None*)

Client connection to the MAL API. This class is used to interact with the API.

Parameters

- **session** (*Optional[Union[aiohttp.ClientSession, asks.Session]]*)
– The session to use for aiohttp/asks requests.

Defaults to creating a new one.

- **lib** (*Optional[str]*) – The async library to use Tokage with. Defaults to *asyncio*. Valid libraries: *asyncio*, *multio*.

- **loop** (*Optional[asyncio.BaseEventLoop]*) – For use with *asyncio*. The event loop to use for *aiohttp*.

Defaults to creating a new one.

session

Union[aiohttp.ClientSession, asks.Session] – The session used for aiohttp/asks HTTP requests.

coroutine **get_anime** (*target_id*)

Retrieves an *Anime* object from an ID

Raises a *AnimeNotFound* Error if an Anime was not found corresponding to the ID.

coroutine **get_character** (*target_id*)

Retrieves a *Character* object from an ID

Raises a *CharacterNotFound* Error if a Character was not found corresponding to the ID.

coroutine **get_manga** (*target_id*)

Retrieves a *Manga* object from an ID

Raises a *MangaNotFound* Error if a Manga was not found corresponding to the ID.

coroutine **get_person** (*target_id*)

Retrieves a *Person* object from an ID

Raises a *PersonNotFound* Error if a Person was not found corresponding to the ID.

coroutine **search_anime** (*query*)

Search for *PartialAnime* by query.

Returns a list of results.

coroutine **search_character** (*query*)

Search for *PartialCharacter* by query.

Returns a list of results.

coroutine **search_id** (*type_, query*)

Parse a google query and return the ID.

Raises a *TokageNotFound* Error if an ID was not found.

coroutine **search_manga** (*query*)

Search for *PartialManga* by query.

Returns a list of results.

coroutine search_person(*query*)
Search for *PartialPerson* by query.
Returns a list of results.

1.2.2 Base Classes

Warning: Do not create these yourself. You'll receive them by way of getter functions.

Anime

class tokage.**Anime**(*anime_id*, *data*, ***kwargs*)
Represents a MAL Anime

id
int – The Anime's ID.

title
str – The Series title.

type
str – The Anime's type. Can be *ONA/OVA/TV/Movie*.

synonyms
list[str] – Alternative names for the Anime.

image
str – The cover image URL for the Anime.

japanese_title
str – Japanese title of the Anime.

status
str – Airing status of the Manga.

episodes
int – Episode count of the Manga.

air_start
str – Airing start date.

air_end
str – Airing end date.

airing
bool – True if the Anime is airing, False if not.

synopsis
str – Description of the Anime.

producers
list[list] – WIP - List of the Anime's producers.

licensors
list[list] – WIP - List of the Anime's licensors.

studios
list[list] – WIP - List of the Anime's studios

premiered*str* – Premier season.**broadcast***str* – Broadcast times.**genres***list[str]* – List of the Anime’s genres.**link***str* – Link to the Anime on MAL.**score***tuple(int)* – Tuple of (score, voters).**duration***str* – Duration of the Anime (may be per episode).**rank***int* – Anime’s rank on the MAL board.**popularity***int* – Popularity rank of the Anime.**members***int* – Amount of members which have the Anime in their list.**favorites***int* – Amount of favorites given to the Anime.**source***str* – Type of source material. Can be *Manga Novel* or *Original*.**related***list[PartialAnime or PartialManga]* – List of related Anime or Manga.

Manga

class tokage.**Manga** (*manga_id*, *data*, ***kwargs*)

Represents a MAL Manga (Includes Novels)

id*int* – The Manga’s ID.**title***str* – The Series title.**type***str* – The Manga’s type. Can be either “Novel” or “Manga”.**synonyms***list[str]* – Alternative names for the Manga.**image***str* – The cover image URL for the Manga.**japanese_title***str* – Japanese title of the Manga.**status***str* – Publishing status of the Manga.

volumes

int – Volume count of the Manga.

chapters

int – Chapter count of the Manga.

publish_start

str – Publication start date.

publish_end

str – Publication end date.

publishing

bool – True if the manga is publishing, False if not.

synopsis

str – Description of the Manga.

author

PartialPerson – PartialPerson instance of the Manga author.

serialization

str – The Manga's serialization.

genres

list[str] – List of the Manga's genres.

link

str – Link to the Manga on MAL.

score

tuple(int) – Tuple of (score, voters).

rank

int – Manga's rank on the MAL board.

popularity

int – Popularity rank of the Manga.

members

int – Amount of members which have the Manga in their list.

favorites

int – Amount of favorites given to the Manga.

related

list[PartialAnime or PartialManga] – List of related Anime or Manga.

Character

class tokage.Character(*char_id*, *data*, ***kwargs*)

Represents a MAL Character

id

int – The Character's ID.

name

str – Character's name.

link

str – Link to the Character on MAL.

image

str – Image URL of the Character.

favorites

int – Amount of favorites the Character has.

animeography

list[*PartialAnime*] – Anime the Character is featured in.

mangaography

list[*PartialManga*] – Manga the Character is featured in.

japanese_name

str – Japanese name of the character.

about

str – WIP - Information about the character. As of now, spoilers are unformatted and will appear.

voice_actors

list[*PartialPerson*] – List of voice actors who played this Character.

Person

class tokage.**Person** (*person_id*, *data*, ***kwargs*)

Represents a MAL Person (Voice Actors, Staff, etc.)

id

int – The Person's ID.

name

str – The Person's name.

link

str – Link to the Person on MAL.

image

str – Image URL of the Person.

favorites

int – Amount of favorites the Person has.

anime

list[*PartialAnime*] – Staff positions in Anime.

manga

list[*PartialManga*] – Published Manga.

more

str – Additional info about the Person.

website

str – Link to the Person's website

voice_acting

list[*PartialCharacter*] – List of characters the Person has voice acted.

1.2.3 Partial Classes

PartialAnime

class tokage.**PartialAnime** (*title, id, url, **kwargs*)

Represents a part of an Anime object

title

str – The Anime’s title.

id

int – The Anime’s ID.

url

str – Link to the Anime.

relation

Optional[str] – relation of the anime to a *Person* or an *Anime*.

coroutine request_full ()

Request an instance of the full, non-partial class. For example, *PartialAnime* -> *Anime*

PartialManga

class tokage.**PartialManga** (*title, id, url, **kwargs*)

Represents a part of a Manga object

title

str – The Manga’s title.

id

int – The Manga’s ID.

url

str – Link to the Manga.

relation

Optional[str] – relation of the manga to a *Person* or a *Manga*.

coroutine request_full ()

Request an instance of the full, non-partial class. For example, *PartialAnime* -> *Anime*

PartialCharacter

class tokage.**PartialCharacter** (*name, id, url, **kwargs*)

Represents a part of a Character object

name

str – The Character’s name.

id

int – The Character’s ID.

url

str – Link to the Character.

anime

Optional[PartialAnime] – The anime this character is from.

```
coroutine request_full()
```

Request an instance of the full, non-partial class. For example, *PartialAnime* -> *Anime*

PartialPerson

```
class tokage.PartialPerson(name, id, url, **kwargs)
```

Represents a part of a Person object

name

str – The Person’s name.

id

int – The Person’s ID.

url

str – Link to the Person.

language

Optional[str] – If this is a partial voice actor, the language of the voice acting.

```
coroutine request_full()
```

Request an instance of the full, non-partial class. For example, *PartialAnime* -> *Anime*

1.3 Exceptions

1.3.1 Generic

```
exception tokage.TokageNotFound
```

This Error is the base class of all Errors in Tokage. Usually you wont recive this error, but some functions may raise it.

1.3.2 Specific

```
exception tokage.AnimeNotFound
```

This Error is raised when an Anime was not found.

```
exception tokage.MangaNotFound
```

This Error is raised when a Manga was not found.

```
exception tokage.CharacterNotFound
```

This Error is raised when a Character was not found.

```
exception tokage.PersonNotFound
```

This Error is raised when a Person was not found.

CHAPTER 2

Indices and tables

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

A

about (tokage.Character attribute), 9
air_end (tokage.Anime attribute), 6
air_start (tokage.Anime attribute), 6
airing (tokage.Anime attribute), 6
Anime (class in tokage), 6
anime (tokage.PartialCharacter attribute), 10
anime (tokage.Person attribute), 9
AnimeNotFound, 11
animeography (tokage.Character attribute), 9
author (tokage.Manga attribute), 8

B

broadcast (tokage.Anime attribute), 7

C

chapters (tokage.Manga attribute), 8
Character (class in tokage), 8
CharacterNotFound, 11
Client (class in tokage), 5

D

duration (tokage.Anime attribute), 7

E

episodes (tokage.Anime attribute), 6

F

favorites (tokage.Anime attribute), 7
favorites (tokage.Character attribute), 9
favorites (tokage.Manga attribute), 8
favorites (tokage.Person attribute), 9

G

genres (tokage.Anime attribute), 7
genres (tokage.Manga attribute), 8
get_anime() (tokage.Client method), 5
get_character() (tokage.Client method), 5
get_manga() (tokage.Client method), 5

get_person() (tokage.Client method), 5

I

id (tokage.Anime attribute), 6
id (tokage.Character attribute), 8
id (tokage.Manga attribute), 7
id (tokage.PartialAnime attribute), 10
id (tokage.PartialCharacter attribute), 10
id (tokage.PartialManga attribute), 10
id (tokage.PartialPerson attribute), 11
id (tokage.Person attribute), 9
image (tokage.Anime attribute), 6
image (tokage.Character attribute), 8
image (tokage.Manga attribute), 7
image (tokage.Person attribute), 9

J

japanese_name (tokage.Character attribute), 9
japanese_title (tokage.Anime attribute), 6
japanese_title (tokage.Manga attribute), 7

L

language (tokage.PartialPerson attribute), 11
licensors (tokage.Anime attribute), 6
link (tokage.Anime attribute), 7
link (tokage.Character attribute), 8
link (tokage.Manga attribute), 8
link (tokage.Person attribute), 9

M

Manga (class in tokage), 7
manga (tokage.Person attribute), 9
MangaNotFound, 11
mangaography (tokage.Character attribute), 9
members (tokage.Anime attribute), 7
members (tokage.Manga attribute), 8
more (tokage.Person attribute), 9

N

name (tokage.Character attribute), 8

name (tokage.PartialCharacter attribute), 10
name (tokage.PartialPerson attribute), 11
name (tokage.Person attribute), 9

P

PartialAnime (class in tokage), 10
PartialCharacter (class in tokage), 10
PartialManga (class in tokage), 10
PartialPerson (class in tokage), 11
Person (class in tokage), 9
PersonNotFound, 11
popularity (tokage.Anime attribute), 7
popularity (tokage.Manga attribute), 8
premiered (tokage.Anime attribute), 6
producers (tokage.Anime attribute), 6
publish_end (tokage.Manga attribute), 8
publish_start (tokage.Manga attribute), 8
publishing (tokage.Manga attribute), 8

R

rank (tokage.Anime attribute), 7
rank (tokage.Manga attribute), 8
related (tokage.Anime attribute), 7
related (tokage.Manga attribute), 8
relation (tokage.PartialAnime attribute), 10
relation (tokage.PartialManga attribute), 10
request_full() (tokage.PartialAnime method), 10
request_full() (tokage.PartialCharacter method), 10
request_full() (tokage.PartialManga method), 10
request_full() (tokage.PartialPerson method), 11

S

score (tokage.Anime attribute), 7
score (tokage.Manga attribute), 8
search_anime() (tokage.Client method), 5
search_character() (tokage.Client method), 5
search_id() (tokage.Client method), 5
search_manga() (tokage.Client method), 5
search_person() (tokage.Client method), 5
serialization (tokage.Manga attribute), 8
session (tokage.Client attribute), 5
source (tokage.Anime attribute), 7
status (tokage.Anime attribute), 6
status (tokage.Manga attribute), 7
studios (tokage.Anime attribute), 6
synonyms (tokage.Anime attribute), 6
synonyms (tokage.Manga attribute), 7
synopsis (tokage.Anime attribute), 6
synopsis (tokage.Manga attribute), 8

T

title (tokage.Anime attribute), 6
title (tokage.Manga attribute), 7

title (tokage.PartialAnime attribute), 10
title (tokage.PartialManga attribute), 10
TokageNotFound, 11
type (tokage.Anime attribute), 6
type (tokage.Manga attribute), 7

U

url (tokage.PartialAnime attribute), 10
url (tokage.PartialCharacter attribute), 10
url (tokage.PartialManga attribute), 10
url (tokage.PartialPerson attribute), 11

V

voice_acting (tokage.Person attribute), 9
voice_actors (tokage.Character attribute), 9
volumes (tokage.Manga attribute), 7

W

website (tokage.Person attribute), 9