
aio-videoindexer

Release 1.0.0.dev3

Chris Lloyd-Jones

Jan 21, 2022

CONTENTS:

1	Overview	1
2	Setup Example	3
3	Main Class	5
	Python Module Index	7
	Index	9

OVERVIEW

An async video indexer package for querying [Microsoft Media Services Video Indexer](#) in Python.

See also:

See the [original application](#) this package was forked from.

1. Install the package from pip with `pip install aio-videoindexer`
2. Subscribe to the video indexer API: <https://api-portal.videoindexer.ai/>
3. Log into your video indexer account: <https://www.videoindexer.ai/account/login>

<p>Warning: This package is in active development, and documentation and examples are missing throughout the application.</p>
--

SETUP EXAMPLE

Here is an example of the package in use:

```
from asyncvideoindexer import AsyncVideoIndexer

VIDEO_INDEXER_ACCOUNT_ID = "your-account-id"
VIDEO_INDEXER_KEY = "your-account-key"
VIDEO_INDEXER_ACCOUNT_LOCATION = "your-account-location"

async def get_video_indexer():
    video_indexer = await AsyncVideoIndexer.AsyncVideoIndexer.create(
        VIDEO_INDEXER_ACCOUNT_ID,
        VIDEO_INDEXER_KEY,
        VIDEO_INDEXER_ACCOUNT_LOCATION,
    )
```


MAIN CLASS

A class for interacting with Video Indexer using asyncio. Does not implement all API facilities.

class AsyncVideoIndexer.AsyncVideoIndexer

A class for interacting with Video Indexer using asyncio.

async classmethod create (*account_id*, *subscription_key*, *location*,
get_access_token_on_startup=True)

Acts as the `__init__` method. This triggers W0201. Using an `@classmethod` rather than `__init__` in order to init async.

async get_access_token () → None

Rotates the current bearer token.

async get_thumbnail (*video_id*, *thumbnail_id*)

Returns the thumbnail content for the provided video ID.

async get_video_access_token (*video_id*, *allow_edit=False*)

Gets a bearer token for accessing an individual video.

async get_video_id_by_external_id (*external_id*)

Gets the Video Indexer ID using an externally provided ID.

async get_video_index (*video_id*)

Returns video information the provided video id.

async get_video_insights_widget_url (*video_id*, *video_access_token*, *allow_edit=False*)

Gets the video insights Widget, as well as whether users should be able to edit.

async get_video_player_widget_url (*video_id*, *video_access_token*)

Gets the video player widget.

async list_videos ()

Lists all videos on the current account.

async upload_video_from_url (*video_name*, *video_external_id*, *callback_url*, *video_url*)

Instructs Video Indexer to ingest a video from provided URL.

async video_indexer_api_request (*api_resource*, *operation*, *params=None*, *headers=None*)

Used for most Async Get and Post API requests.

async video_indexer_auth_request (*api_resource*, *params=None*)

Used to get Access tokens- generally and for video access

PYTHON MODULE INDEX

a

`AsyncVideoIndexer`, [5](#)

INDEX

A

`AsyncVideoIndexer`
module, 5

`AsyncVideoIndexer` (class in `AsyncVideoIndexer`), 5

C

`create()` (`AsyncVideoIndexer.AsyncVideoIndexer`
class method), 5

G

`get_access_token()` (`AsyncVideoIndexer.AsyncVideoIndexer`
method), 5

`get_thumbnail()` (`AsyncVideoIndexer.AsyncVideoIndexer`
method), 5

`get_video_access_token()` (`AsyncVideoIndexer.AsyncVideoIndexer`
method), 5

`get_video_id_by_external_id()`
(`AsyncVideoIndexer.AsyncVideoIndexer`
method), 5

`get_video_index()` (`AsyncVideoIndexer.AsyncVideoIndexer`
method), 5

`get_video_insights_widget_url()`
(`AsyncVideoIndexer.AsyncVideoIndexer`
method), 5

`get_video_player_widget_url()`
(`AsyncVideoIndexer.AsyncVideoIndexer`
method), 5

L

`list_videos()` (`AsyncVideoIndexer.AsyncVideoIndexer`
method), 5

M

module
`AsyncVideoIndexer`, 5

U

`upload_video_from_url()` (`AsyncVideoIndexer.AsyncVideoIndexer`
method), 5

V

`video_indexer_api_request()` (`AsyncVideoIndexer.AsyncVideoIndexer`
method), 5

`video_indexer_auth_request()`
(`AsyncVideoIndexer.AsyncVideoIndexer`
method), 5