
Flask-arango-orm Documentation

Release 0.1.1

Caitlyn O'Hanna

Sep 24, 2019

Contents:

1	Configuration Options	1
2	API	3
3	About	5
4	Installation	7
5	Usage	9
6	Indices and tables	11
	Python Module Index	13
	Index	15

Configuration Options

All configuration options are to be specified in the Flask app's config

ARANGODB_DATABASE

str

Used to specify the database name used to connect to arangodb

ARANGODB_USER

str

Used to specify username used to connect to arangodb

ARANGODB_PASSWORD

str

Used to specify password used to connect to arangodb

ARANGODB_HOST

tuple

Default is ('http', '127.0.0.1', 8529)

A tuple in the format of *protocol, host, port* Used for connecting to the arangodb host

ARANGODB_CLUSTER

bool

Default is *False*

Used to determine if connecting to a pool of hosts

ARANGODB_HOST_POOL

list

A list of tuples in the format of *protocol, host, port* Used for connecting to a pool of arangodb hosts

Main module for flask_arango_orm

class flask_arango_orm.arango.**ArangoORM** (*app=None*)
Flask extension for integrating the arango_orm package

__init__ (*app=None*)
Constructor

If a flask app instance is passed as an argument, *init_app* is run as well.

Parameters *app* (*flask.Flask*) – Flask app instance

__weakref__
list of weak references to the object (if defined)

connect ()
Sets up the connection to the arangodb database

Uses app configuration for setup, utilizing the following:

- ARANGODB_DATABASE
- ARANGODB_USER
- ARANGODB_PASSWORD
- ARANGODB_HOST
- ARANGODB_CLUSTER
- ARANGODB_HOST_POOL

If ARANGODB_CLUSTER is True, then the host configuration is taken from ARANGODB_HOST_POOL, otherwise ARANGODB_HOST is used.

Returns Connection to arangodb

Return type arango_orm.Database

connection

Property for storing and retrieving the database connection

Stores the database connection in the top of the Flask application context stack, `flask._app_ctx_stack`. If a connection does not currently exist, `connect()` is called.

Returns ArangoDB connection

Return type `arango_orm.Database`

init_app (*app*)

Initializes ArangoORM for use with the passed app instance

Sets up the following defaults:

- `ARANGODB_CLUSTER = False`
- `ARANGODB_HOST = ('http', '127.0.0.1', 8529)`

This should be useful for most development setups

Parameters `app` (*flask.Flask*) – Flask app instance

Returns

CHAPTER 3

About

Flask-arango-orm is used to connect to an [ArangoDB](#) instance using [arango-orm](#) as an object model layer to your [Flask](#) app. ArangoDB is a hybrid database that can provide a document, graph, and relational mode which can be taken advantage of using arango-orm.

CHAPTER 4

Installation

Using pip:

```
pip install Flask-arango-orm
```

Or using setup.py:

```
python setup.py install
```

Tests can be ran using:

```
python setup.py test
```

Documentation can be generated using:

```
python setup.py build_sphinx
```


This extension for the Flask framework uses arango-orm to provide an Object Model to use within the Flask application.

```
from flask import Flask
from flask_arango_orm import ArangoORM

app = Flask(__name__)
arango = ArangoORM(app)

@app.route('/route')
def some_route():
    db_conn = arango.connection
```


CHAPTER 6

Indices and tables

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

f

`flask_arango_orm`, 3

`flask_arango_orm.arango`, 3

Symbols

`__init__()` (*flask_arango_orm.arango.ArangoORM* method), 3

`__weakref__` (*flask_arango_orm.arango.ArangoORM* attribute), 3

A

ARANGODB_CLUSTER, 1

ARANGODB_DATABASE, 1

ARANGODB_HOST, 1

ARANGODB_HOST_POOL, 1

ARANGODB_PASSWORD, 1

ARANGODB_USER, 1

ArangoORM (*class in flask_arango_orm.arango*), 3

B

bool, 1

C

`connect()` (*flask_arango_orm.arango.ArangoORM* method), 3

`connection` (*flask_arango_orm.arango.ArangoORM* attribute), 3

F

`flask_arango_orm` (*module*), 3

`flask_arango_orm.arango` (*module*), 3

I

`init_app()` (*flask_arango_orm.arango.ArangoORM* method), 4

L

list, 1

S

str, 1

T

tuple, 1