<https://realpython.com/introduction-to-mongodb-and-python/>

$ pip install pymongo==3.11.2

>>> import pymongo

PyMongo -drivers

<https://pymongo.readthedocs.io/en/stable/api/pymongo/index.html>

Database:

<https://pymongo.readthedocs.io/en/stable/api/pymongo/database.html>

<https://pymongo.readthedocs.io/en/stable/api/pymongo/collection.html>

Establishing connection:

>>> from pymongo import MongoClient

>>> client = MongoClient()

>>> client

MongoClient(host=['localhost:27017'], ..., connect=True)

Creatte DB:

>>> db = client.rptutorials

>>> db

Database(MongoClient(host=['localhost:27017'], ..., connect=True), 'rptutorials')

only close it before exiting the application to clear all the acquired resources.

>>> client.close()

To retrieve documents from a collection, you can use [.find()](https://pymongo.readthedocs.io/en/stable/api/pymongo/collection.html#pymongo.collection.Collection.find).

>>> import pprint

>>> for doc in tutorial.find():

... pprint.pprint(doc)

...

{'\_id': ObjectId('600747355e6ea8d224f754ba'),

'author': 'Jon',

'contributors': ['Aldren', 'Geir Arne', 'Joanna', 'Jason'],

'title': 'Reading and Writing CSV Files in Python',

'url': 'https://realpython.com/python-csv/'}

You can also use .find\_one() to retrieve a single document.

>>> jon\_tutorial = tutorial.find\_one({"author": "Jon"})

PyMongo also provides methods to [replace](https://pymongo.readthedocs.io/en/stable/api/pymongo/collection.html#pymongo.collection.Collection.replace_one), [update](https://pymongo.readthedocs.io/en/stable/api/pymongo/collection.html#pymongo.collection.Collection.update_one), and [delete](https://pymongo.readthedocs.io/en/stable/api/pymongo/collection.html#pymongo.collection.Collection.delete_one) documents from a database. If you want to dive deeper into these features, then take a look at the [documentation](https://pymongo.readthedocs.io/en/stable/api/pymongo/collection.html) for Collection.

Mango engine

$ pip install mongoengine==0.22.1

>>> from mongoengine import Document, ListField, StringField, URLField

>>> class Tutorial(Document):

... title = StringField(required=True, max\_length=70)

... author = StringField(required=True, max\_length=20)

... contributors = ListField(StringField(max\_length=20))

... url = URLField(required=True)

Inserting

>>> tutorial1.save() # Insert the new tutorial

Print title data to the screen

>>> for doc in Tutorial.objects(author="Alex"):

... print(doc.title)

...

Python's Requests Library (Guide)

Some more examples of PyMongo:

<https://www.bogotobogo.com/python/MongoDB_PyMongo/python_MongoDB_pyMongo_tutorial_installing.php>

def get\_db():

from pymongo import MongoClient

client = MongoClient('localhost:27017')

db = client.myFirstMB

return db

def add\_country(db):

db.countries.insert({"name" : "Canada"})

def get\_country(db):

return db.countries.find\_one()

if \_\_name\_\_ == "\_\_main\_\_":

db = get\_db()

add\_country(db)

print get\_country(db)