**Import gói pymongo để làm việc trong python:**

Import pymongo

Hoặc cho tiện

From pymongo import MongoClient

**Kết nối với csdl trước trong hàm \_\_init\_\_:**

Def \_\_init\_\_(self):

Client = MongoClient() (tạo kết nối connection tên client)

Db = client.test-database (truy cập 1 database cụ thể)

Hoặc db = client[‘test-database’]

Collection = db.test-collection (truy cập collection tương đương table-bảng)

Hoặc collect = db[‘test-collection’]

**Documents (hàng trong bảng dữ liệu-Collection)**

**Insert 1 document vào collection, sử dụng insert\_one(post)**

**>>> import** **datetime**

**>>>** post = {"author": "Mike",

**...**  "text": "My first blog post!",

**...**  "tags": ["mongodb", "python", "pymongo"],

**...**  "date": datetime.datetime.utcnow()}

**>>>** posts = db.posts

**>>>** post\_id = posts.insert\_one(post).inserted\_id

**>>>** post\_id

ObjectId('59db233add1f783a2073ccc5')

**Xem các collection trong database:**

>>> db.collection\_names()

['test-collection']

>>> db.collection\_names(include\_system\_collections=False)

['test-collection']

**Lấy document-row đầu tiên trong collection-table:**

>>> collect.find\_one()

{'text': 'My first document in mongodb', 'author': 'me', '\_id': ObjectId('59db233add1f783a2073ccc5'), 'date': datetime.datetime(2017, 10, 9, 7, 19, 11, 790000), 'tags': ['mongodb', 'python', 'pymongo']}

>>> import pprint (hiển thị dữ liệu dễ nhìn hơn)

>>> pprint.pprint(collect.find\_one())

{'\_id': ObjectId('59db233add1f783a2073ccc5'),

'author': 'me',

'date': datetime.datetime(2017, 10, 9, 7, 19, 11, 790000),

'tags': ['mongodb', 'python', 'pymongo'],

'text': 'My first document in mongodb'}

>>> doc = collect.find\_one()['\_id'] (trường \_id có kiểu dữ liệu là BSON-binary json, ko rõ)

>>> doc

*ObjectId('59db233add1f783a2073ccc5')*

>>> print (doc)

59db233add1f783a2073ccc5

**Tìm theo 1 query :**

>>> pprint.pprint(collect.find\_one({"author" : "me"}))

{'\_id': ObjectId('59db233add1f783a2073ccc5'),

'author': 'me',

'date': datetime.datetime(2017, 10, 9, 7, 19, 11, 790000),

'tags': ['mongodb', 'python', 'pymongo'],

'text': 'My first document in mongodb'}

**Ko tìm thấy sẽ trả vè None:**

>>> pprint.pprint(collect.find\_one({"author" : "me1"}))

None

**Tìm document theo ObjectId:**

>>> pprint.pprint(collect.find\_one({"id":post\_id})) (**chú ý là \_id cơ**)

None

>>> pprint.pprint(collect.find\_one({"\_id":post\_id}))

{'\_id': ObjectId('59db233add1f783a2073ccc5'),

'author': 'me',

'date': datetime.datetime(2017, 10, 9, 7, 19, 11, 790000),

'tags': ['mongodb', 'python', 'pymongo'],

'text': 'My first document in mongodb'}

**ObjectId khác với biểu diễn string của nó:**

>>> post\_id\_as\_str = str(post\_id)

>>> post\_id\_as\_str

'59db233add1f783a2073ccc5'

>>> collect.find\_one({"\_id":post\_id\_as\_str})

>>> result = collect.find\_one({"\_id":post\_id\_as\_str}) #No result (ko có kq)

>>> result

>>>

**Chuyển string của ObjectId sang kiểu ObjectId trước rồi mới query:**

>>> **from bson.objectid import ObjectId**

>>> doc = collect.find\_one({"\_id":ObjectId(post\_id)})

>>> doc

{'text': 'My first document in mongodb', 'author': 'me', '\_id': ObjectId('59db233add1f783a2073ccc5'), 'date': datetime.datetime(2017, 10, 9, 7, 19, 11, 790000), 'tags': ['mongodb', 'python', 'pymongo']}

Chú ý vè Unicode Strings:

Dữ liệu được crawl từ web toàn có dạng: u’Hello thay vì ‘Hello’. Vì:

Nó ( u’Hello’ ) là định dạng kiểu **Unicode strings (<type ‘unicode’>)** còn hiển thị là ‘Hello’ thì nó là có định dạng kiểu **Regular strings (<type ‘str’>)**

MongoDB stores data in [BSON format](http://bsonspec.org/). BSON strings are UTF-8 encoded so PyMongo must ensure that any strings it stores contain only valid UTF-8 data. Regular strings (<type ‘str’>) are validated and stored unaltered. Unicode strings (<type ‘unicode’>) are encoded UTF-8 first. The reason our example string is represented in the Python shell as u’Mike’ instead of ‘Mike’ is that PyMongo decodes each BSON string to a Python unicode string, not a regular str.

(Mongo DB lưu dl ở định dạng BSON. BSON strings là được mã hóa với định dạng UTF-8 nên PyMongo phải đảm bảo bất kì strings nó lưu chỉ cho phép kiểu dl hợp lệ là UTF-8. String bình thường (<type ‘str’>) là đã được coi là hợp lệ và lưu bình thường ko cần thay đổi-chuyển kiểu. Unicode string (<type ‘unicode’>) sẽ được chuyển sang UTF-8 trước(2 kiểu khác nhau nhá).

Lí do trong vd lại hiển thị trong Python shell là u’Mike’ là vì PyMongo giải mã mỗi BSON string sang Python Unicode string – ko phải 1 tring kiểu str thông thường.

**Thêm nhiều document vào collection, sử dụng insert\_many(list):**

**>>>** new\_posts = [{"author": "Mike",

**...**  "text": "Another post!",

**...**  "tags": ["bulk", "insert"],

**...**  "date": datetime.datetime(2009, 11, 12, 11, 14)},

**...**  {"author": "Eliot",

**...**  "title": "MongoDB is fun",

**...**  "text": "and pretty easy too!",

**...**  "date": datetime.datetime(2009, 11, 10, 10, 45)}]

**>>>** result = posts.insert\_many(new\_posts)

**>>>** result.inserted\_ids

[ObjectId('...'), ObjectId('...')]

**Chú ý : thấy rằng phần tử thứ 2 new\_posts[1] có shape-hình dạng khác với cái kia, có thêm 1 trường title và thiếu trường tags. Vẫn insert được bình thường MongoDB:schema-free**

(trong csdl nó tự thêm 1 trường title mới – hay vler)

**Lấy hơn 1 document dùng hàm collection.find() -> trả về 1 Cursor(con trỏ) rồi dùng for lặp:**

**>>> for** post **in** posts.find():

**...**  pprint.pprint(post)

**...**

{u'\_id': ObjectId('...'),

u'author': u'Mike',

u'date': datetime.datetime(...),

u'tags': [u'mongodb', u'python', u'pymongo'],

u'text': u'My first blog post!'}

{u'\_id': ObjectId('...'),

u'author': u'Mike',

u'date': datetime.datetime(...),

u'tags': [u'bulk', u'insert'],

u'text': u'Another post!'}

hoặc kèm theo query:

**>>> for** post **in** posts.find({"author": "Mike"}):

**...**  pprint.pprint(post)

**Đếm số document lấy được:**

**>>>** posts.count()

3

Hoặc

**>>>** posts.find({"author": "Mike"}).count()

2

Truy vấn khoảng(range):

**>>>** d = datetime.datetime(2009, 11, 12, 12)

**>>> for** post **in** posts.find({"date": **{"$lt": d}**}).sort("author"):

**...**  pprint.pprint(post)

**...**

{u'\_id': ObjectId('...'),

u'author': u'Eliot',

u'date': datetime.datetime(...),

u'text': u'and pretty easy too!',

u'title': u'MongoDB is fun'}

{u'\_id': ObjectId('...'),

u'author': u'Mike',

u'date': datetime.datetime(...),

u'tags': [u'bulk', u'insert'],

u'text': u'Another post!'}

tìm các post trước ngày d ($lt – nhỏ hơn)

**Các truy vấn nâng cao hơn:**

<https://docs.mongodb.com/manual/reference/operator/>

Các chỉ mục duy nhất unique: