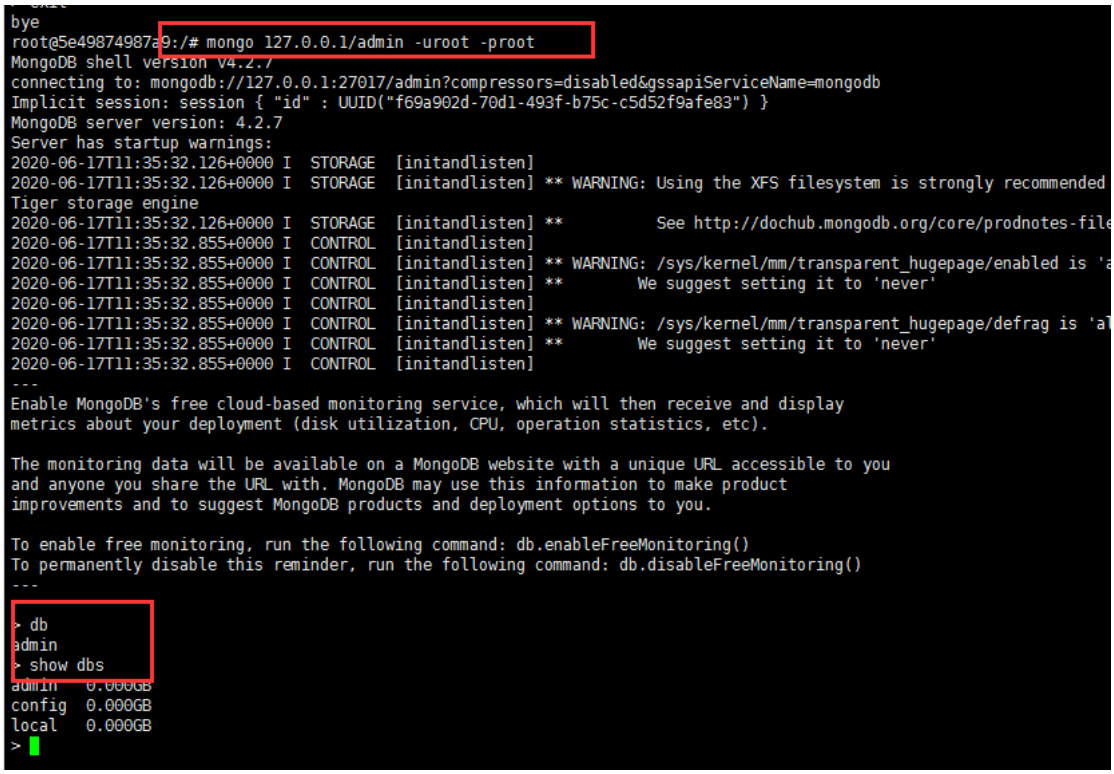
学习目标：MongoDB 的基本操作

Test1 集合



1 Create DataBase

```
> db
admin
> show dbs
admin   0.000GB
config  0.000GB
local   0.000GB
> use test1
switched to db test1
> db
test1
> show dbs
admin   0.000GB
config  0.000GB
local   0.000GB
> db.tes1.insert({"name":"jhtchina"})
WriteResult({ "nInserted" : 1 })
>
```

```
> show dbs
admin   0.000GB
config  0.000GB
local   0.000GB
test1   0.000GB
>
```

2 Drop DataBase

```
> db.drop
db.dropAllRoles(  db.dropAllUsers(  db.dropDatabase(  db.dropRole(      db.dropUser(
> db.dropDatabase()
{ "dropped" : "test1", "ok" : 1 }
> show dbs
admin   0.000GB
config  0.000GB
local   0.000GB
>
```

3 Create Collection(创建集合)

```
> db.createCollection("table1")
{ "ok" : 1 }
```

```
> show collections
table1
tes1
>
```

```
> db.table1.insert({"age":18})
WriteResult({ "nInserted" : 1 })
>
```

4 Drop Collection

```
> db.table1.drop()
true
```

```
> show collections
tes1
```

5 Insert Documents

```
> db.table1.insert({"age":18})
WriteResult({ "nInserted" : 1 })
```

6 Query Documents

```
> db.test1.insert({"age":190})
WriteResult({ "nInserted" : 1 })
> db.test1.insert({"age":191})
WriteResult({ "nInserted" : 1 })
> db.test1.insert({"age":192})
WriteResult({ "nInserted" : 1 })
> db.test1.find({})
{ "_id" : ObjectId("5eea0a18c23a4d1ed1529004"), "age" : 190 }
{ "_id" : ObjectId("5eea0a1dc23a4d1ed1529005"), "age" : 191 }
{ "_id" : ObjectId("5eea0a1fc23a4d1ed1529006"), "age" : 192 }
> db.test1.find({"age":191})
{ "_id" : ObjectId("5eea0a1dc23a4d1ed1529005"), "age" : 191 }
>
```

7 Update Documents

```
> db.test1.update({"age":190},{$set:{"age":1900}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.test1.find({})
{ "_id" : ObjectId("5eea0a18c23a4d1ed1529004"), "age" : 1900 }
{ "_id" : ObjectId("5eea0a1dc23a4d1ed1529005"), "age" : 191 }
{ "_id" : ObjectId("5eea0a1fc23a4d1ed1529006"), "age" : 192 }
>
```

8 Delete Documents

```
> db.test1.remove({"age":1900})
WriteResult({ "nRemoved" : 1 })
> db.test1.find({})
{ "_id" : ObjectId("5eea0a1dc23a4d1ed1529005"), "age" : 191 }
{ "_id" : ObjectId("5eea0a1fc23a4d1ed1529006"), "age" : 192 }
>
```

9 mongodump 备份 mock 库

```
root@546ba38a8003:/home/dump/mock# mongodump -h 127.0.0.1:27017 -uroot -proot --authenticationDatabase admin  -d mock -o /home/
2020-06-19T08:24:23.035+0000    writing mock.orders to
2020-06-19T08:24:23.057+0000    writing mock.fruit to
2020-06-19T08:24:23.059+0000    done dumping mock.fruit (1 document)
2020-06-19T08:24:23.578+0000    done dumping mock.orders (100000 documents)
root@546ba38a8003:/home/dump/mock#
```

10 mongorestore 恢复数据备份

```
root@546ba38a8003:/home/dump# mongorestore /home/dump/mock/orders.bson  -uroot -proot
2020-06-18T05:43:30.241+0000    checking for collection data in /home/dump/mock/orders.bson
2020-06-18T05:43:30.300+0000    reading metadata for mock.orders from /home/dump/mock/orders.metadata.json
2020-06-18T05:43:30.341+0000    restoring mock.orders from /home/dump/mock/orders.bson
2020-06-18T05:43:32.782+0000    no indexes to restore
2020-06-18T05:43:32.782+0000    finished restoring mock.orders (100000 documents, 0 failures)
2020-06-18T05:43:32.782+0000    100000 document(s) restored successfully. 0 document(s) failed to restore.
root@546ba38a8003:/home/dump#
```

mongorestore /home/dump/mock/orders.bson -uroot -proot

```
root@546ba38a8003:/home/dump# mongorestore /home/dump/mock/orders.bson  -uroot -proot
2020-06-18T05:43:30.241+0000    checking for collection data in /home/dump/mock/orders.bson
2020-06-18T05:43:30.300+0000    reading metadata for mock.orders from /home/dump/mock/orders.metadata.json
2020-06-18T05:43:30.341+0000    restoring mock.orders from /home/dump/mock/orders.bson
2020-06-18T05:43:32.782+0000    no indexes to restore
2020-06-18T05:43:32.782+0000    finished restoring mock.orders (100000 documents, 0 failures)
2020-06-18T05:43:32.782+0000    100000 document(s) restored successfully. 0 document(s) failed to restore.
root@546ba38a8003:/home/dump# mongo 127.0.0.1/admin -uroot -proot
MongoDB shell version v4.2.7
connecting to: mongodb://127.0.0.1:27017/admin?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("2e9cda11-5b7c-4278-9a74-45292b824f68") }
MongoDB server version: 4.2.7
Server has startup warnings:
2020-06-18T02:28:38.941+0000 I  STORAGE  [initandlisten]
2020-06-18T02:28:38.941+0000 I  STORAGE  [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTig
2020-06-18T02:28:38.941+0000 I  STORAGE  [initandlisten] **          See http://dochub.mongodb.org/core/prodnotes-filesystem
2020-06-18T02:28:40.079+0000 I  CONTROL  [initandlisten]
2020-06-18T02:28:40.079+0000 I  CONTROL  [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.
2020-06-18T02:28:40.079+0000 I  CONTROL  [initandlisten] **        We suggest setting it to 'never'
2020-06-18T02:28:40.079+0000 I  CONTROL  [initandlisten]
2020-06-18T02:28:40.079+0000 I  CONTROL  [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/defrag is 'always'.
2020-06-18T02:28:40.079+0000 I  CONTROL  [initandlisten] **        We suggest setting it to 'never'
2020-06-18T02:28:40.079+0000 I  CONTROL  [initandlisten]
---
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---

> show dbs
admin   0.000GB
config  0.000GB
db1     0.000GB
local   0.000GB
mock    0.046GB
>
```

```
> show dbs
admin   0.000GB
config  0.000GB
db1     0.000GB
local   0.000GB
mock    0.046GB
> use mock
switched to db mock
> show collections
orders
> db.orders.findOne({})
{
        "_id" : ObjectId("5dbe7a542411dc9de64291ab"),
        "street" : "7563 Thaddeus Courts",
        "city" : "Metzville",
        "state" : "Colorado",
        "country" : "Kazakhstan",
        "zip" : "01028",
        "phone" : "104.627.5710 x005",
        "name" : "Rosemary Kertzmann",
        "userId" : 2131,
        "orderDate" : ISODate("2019-08-17T07:13:35.063Z"),
        "status" : "shipping",
        "shippingFee" : NumberDecimal("7.00"),
        "orderLines" : [
                {
                        "product" : "Generic Metal Pizza",
                        "sku" : "9166",
                        "qty" : 18,
                        "price" : NumberDecimal("32.00"),
                        "cost" : NumberDecimal("29.12")
                },
                {
                        "product" : "Refined Plastic Shirt",
                        "sku" : "5531",
                        "qty" : 100,
                        "price" : NumberDecimal("28.00"),
                        "cost" : NumberDecimal("26.32")
```

docker 操作

(1) docker search mongo

(2) docker pull mongo:latest

(3) docker run -itd --name mongo -p 27017:27017 mongo --auth

    docker run -itd --name mongo_1 -p 27019:27017 mongo --auth

(4) docker stop 容器 ID

docker rm -f 容器 ID

(5) docker exec -it 546ba38a8003 /bin/bash

(6) Mongo

use admin

db.createUser({user:"root",pwd:"root",roles:[{role:'root',db:'admin'}]})

(7) mongo 127.0.0.1/admin -uroot -proot

客户端做一次 telnet ，还有 navicat 做一次连接测试

MongoDB CRUD Operations

(1) use jhtchinadb

db

show dbs

(2) use jhtchinadb

db.dropDatabase()

(3) use jhtchinadb

db.createCollection("table1")

show collections

db.table1.find({})

(4) db.table1.drop()

(5)

db.createCollection("table1")

db.table1.insert({"name":"jhtchina","Age":18,"role":["student","teacher","engineer"]})

db.table1.insert({"name":"test1","Age":20,"role":["baby","doctor","nurse"]})

(6)

db.table1.find({"name":"test1"})

db.table1.find({"name":"jhtchina"})

db.table1.find({"Age":18})

db.table1.find({$or:[{"name":"jhtchina"},{"Age": 20}]}).pretty()

db.table1.find({$or:[{"name":"jhtchina"},{"Age": 20}]})

db.table1.find({$or:[{"name":"jhtchina"},{"Age": 20}]})

db.table1.find({"Age":{$lt:19}}).pretty()

(7)

db.table1.update({'name':'jhtchina'},{$set:{'Age':40}})

(8)

db.table1.remove({"name":"jhtchina"})

//有其他参数，这里不做讲解

(9)

mongodump -h 127.0.0.1:27017 -uroot -proot --authenticationDatabase admin -d jhtchinadb -o /home

(10)

mongo 127.0.0.1/admin -uroot -proot

use jhtchinadb

db.dropDatabase()

show dbs

mongorestore /home/jhtchinadb/table1.bson -uroot -proot