# MongoDB安装

## yum安装

创建yum源文件：

cd /etc/yum.repos.d

vim mongodb-org-4.0.repo

添加以下内容：

[mngodb-org]

name=MongoDB Repository

baseurl=http://mirrors.aliyun.com/mongodb/yum/redhat/7Server/mongodb-org/4.0/x86\_64/

gpgcheck=0

enabled=1

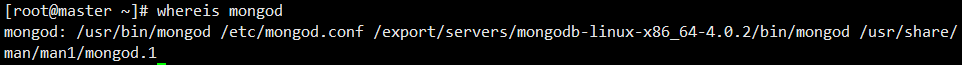
安装MongoDB

安装命令：

yum -y install mongodb-org

安装完成后,查看mongo安装位置

whereis mongod



systemctl start mongod.service

## 解压安装

MongoDB官网 <https://www.mongodb.com>

cd /export/softwares/

wget https://fastdl.mongodb.org/linux/mongodb-linux-x86\_64-4.0.2.tgz

tar zxvf mongodb-linux-x86\_64-4.0.2.tgz -C ../servers/

vi /etc/profile

export MONGODB\_HOME=/export/servers/mongodb-linux-x86\_64-4.0.2

export PATH=:$MONGODB\_HOME/bin:$PATH

cd /export/servers/mongodb-linux-x86\_64-4.0.2

mkdir -p /export/servers/mongodb-linux-x86\_64-4.0.2/db

开启mongodb

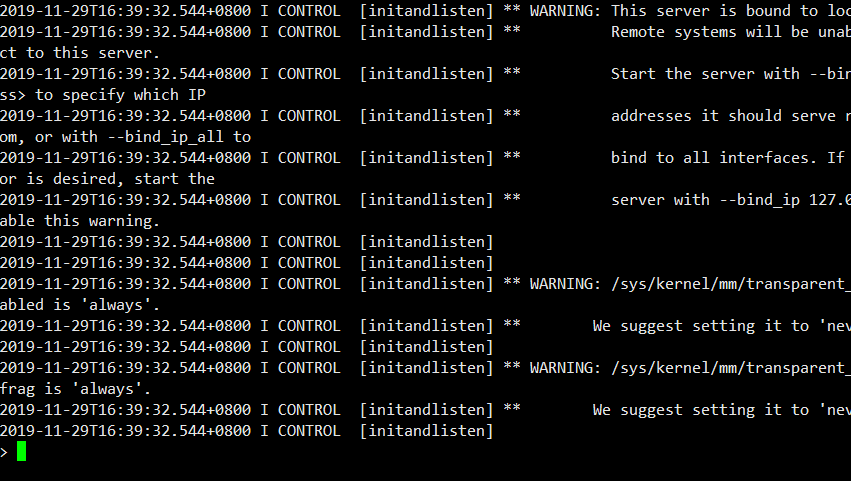
mongod --dbpath /export/servers/mongodb-linux-x86\_64-4.0.2/db

关闭mongodb

mongod -shutdown

进入shell

mongo --host 127.0.0.1:27017



## 配置文件

vi /etc/mongod.conf

bindIp: 0.0.0.0

# shell

mongo

# 数据库和集合操作

## 数据库操作

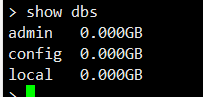
查看当前连接服务器

db.getMongo()



查看数据库列表

show dbs



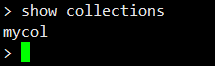
切换数据库

use test



查看数据库中所有集合

show collections



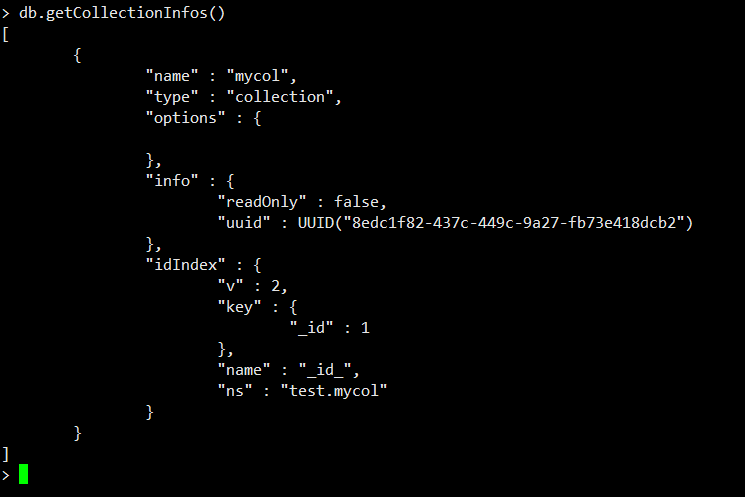
json显示集合名称

db.getCollectionNames()



集合详细信息

db.getCollectionInfos()



显示当前数据库名

db



删除数据库

db.dropDatabase()

## 集合操作

新建集合

db.createCollection("mycol")



删除集合

db.myCol.drop()

## 基本增删改查操作

文档插入

db.mycol.insert({

item1:'111111',

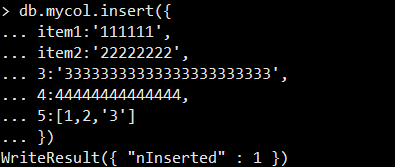
item2:'22222222',

3:'33333333333333333333333',

4:44444444444444,

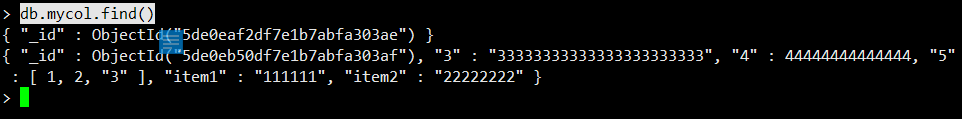
5:[1,2,'3']

})

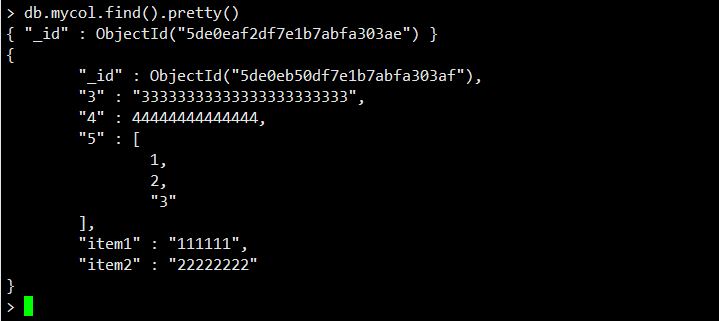


查看文档

db.mycol.find()

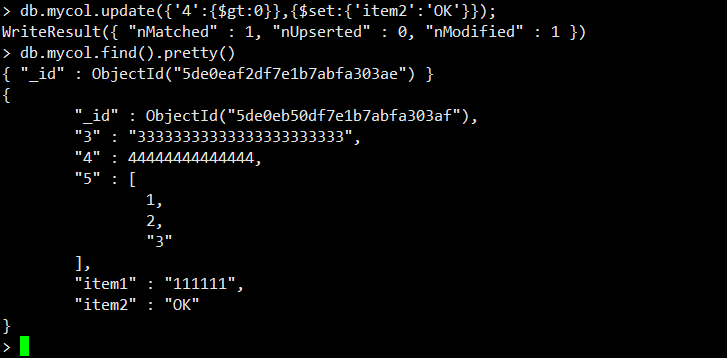


db.mycol.find().pretty()



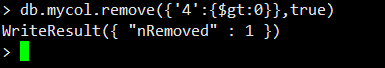
## 文档更新

db.mycol.update({'4':{$gt:0}},{$set:{'item2':'OK'}});



文档删除

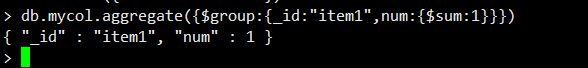
db.mycol.remove({'4':{$gt:0}},true)



db.mycol.remove({})

## 聚合和管道

db.mycol.aggregate({$group:{\_id:"item1",num:{$sum:1}}})

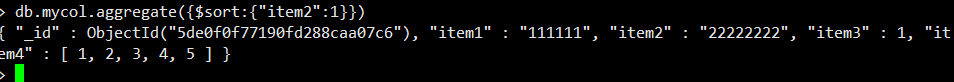


db.mycol.aggregate({$project:{item1:1, item2:1, item3:1}})

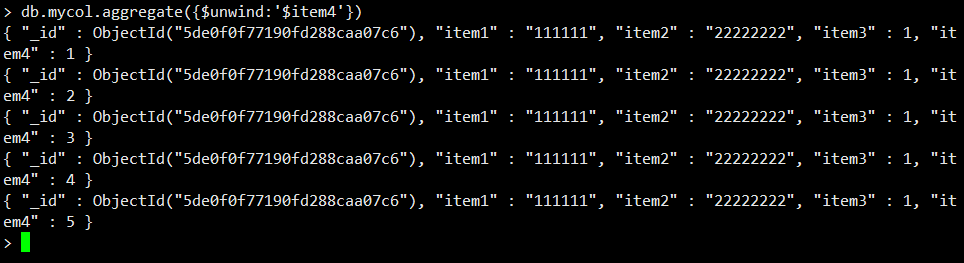


db.mycol.aggregate({$match:{"item3":{$gt:1,$gt:15}}})

db.mycol.aggregate({$sort:{"item2":1}})

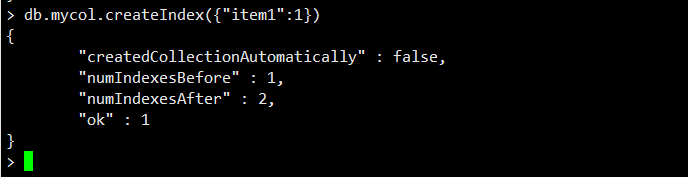


db.mycol.aggregate({$unwind:'$item4'})



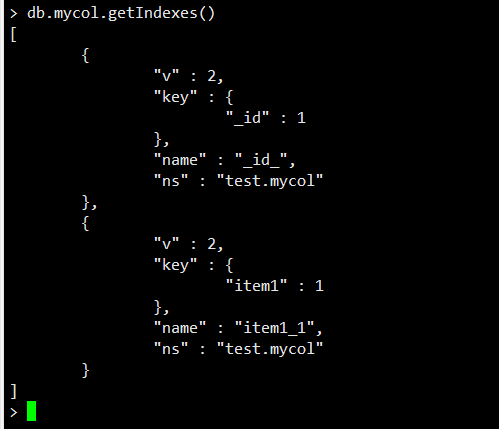
## 索引操作

db.mycol.createIndex({"item1":1})



查看索引

db.mycol.getIndexes()



删除索引

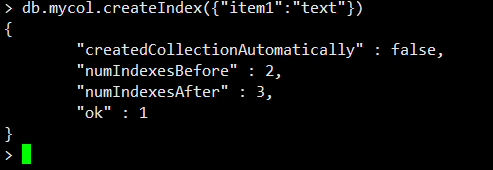
db.mycol.dropIndex("myindex")



db.mycol.dropIndexes()

全文索引

db.mycol.createIndex({"item1":"text"})



# python访问MongoDB

## 安装

pip install pymongo

## 导包

from pymongo import MongoClient

## 建立连接

client=MongoClient("192.168.52.129:27017")

## 切换数据库

db=client.get\_database("testdb")

db=client.testdb

## 切换集合

col=db.testcol

## 定义JSON文档

item={

"name":"fruits",

"count\_of":3,

"varieties":['banana',"cherry","orange"]

}

item1={

"name":"fruits",

"count\_of":4,

"varieties":['banana',"cherry","orange"]

}

item2={

"name":"fruits",

"count\_of":5,

"varieties":['banana',"cherry","orange"]

}

## 插入记录

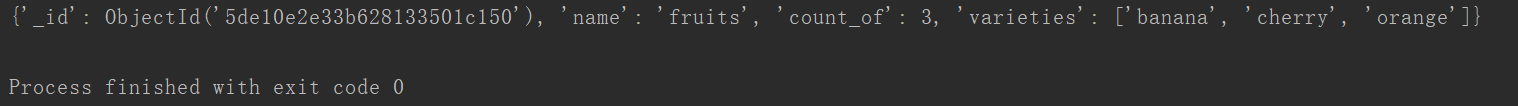
col.insert\_one(item)

col.insert\_one(item1)

col.insert\_one(item2)

## 查看数据

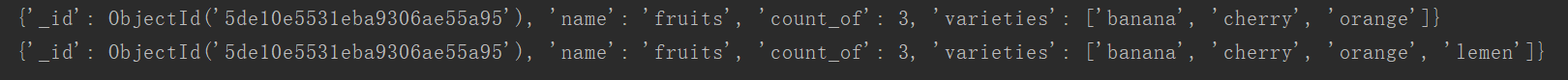
print(col.find\_one())



## 更新数据

col.update\_many({"name":"fruits"},{"$push":{"varieties":"lemen"}})

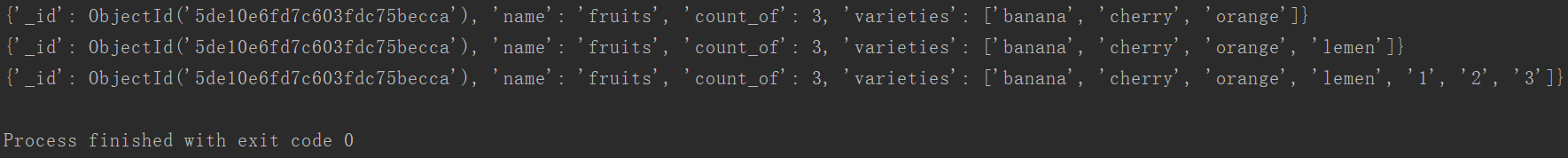
print(col.find\_one())



## $push $each插入多个元素

col.update\_many({"name":"fruits"},{"$push":{"varieties":{"$each":["1",'2','3']}}})

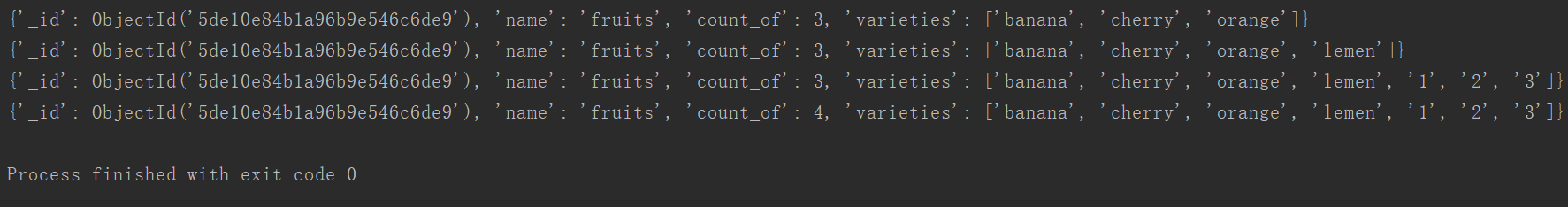
print(col.find\_one())



## 累加方式更新数字类型元素

col.update\_many({"name":"fruits"},{"$inc":{"count\_of":1}})

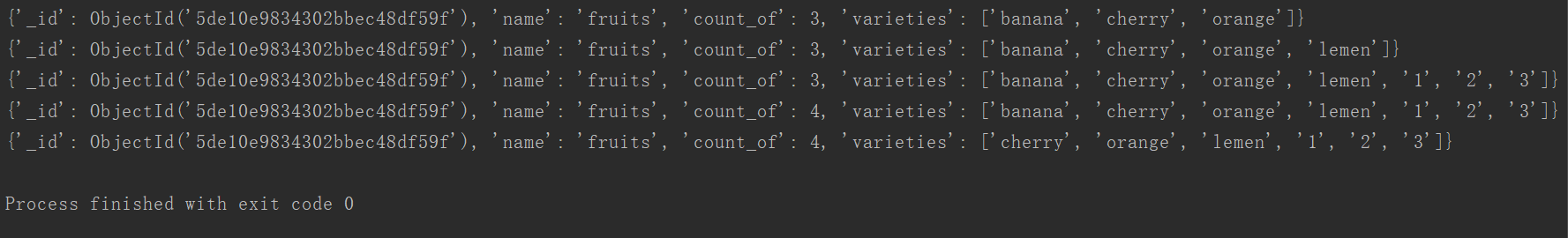
print(col.find\_one())



## $pop删除数据

col.update\_many({"name":"fruits"},{"$pop":{"varieties":-1}})

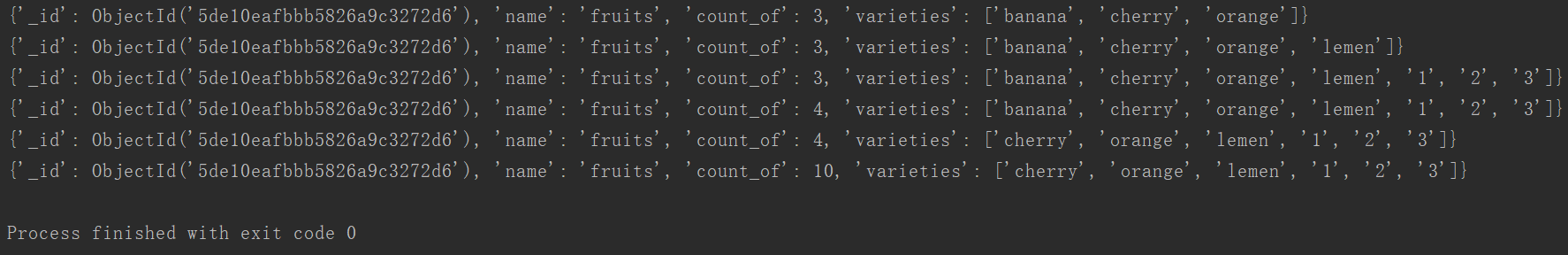
print(col.find\_one())



## 更新数据

col.update\_many({"name":"fruits"},{"$set":{"count\_of":10}})

print(col.find\_one())



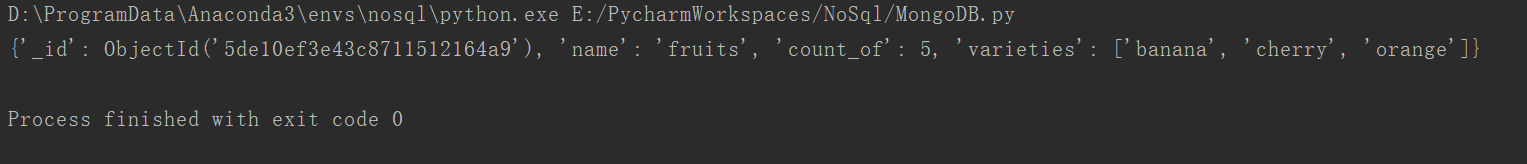
## 查看数据

print(col.find\_one({"name":"fruits"}))

## 排序 限制 跳过

for r in col.find({"name":"fruits"}).sort("count\_of").limit(3).skip(2):

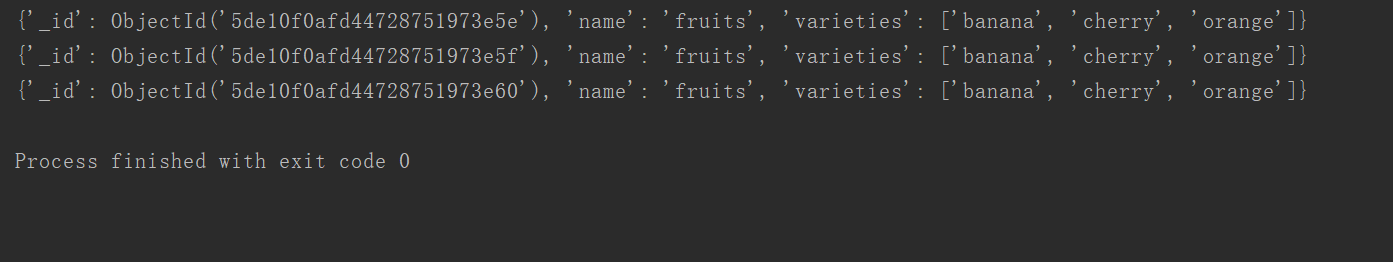
print(r)



## 控制显示的列

for r in col.find({"name":"fruits"},projection={'count\_of':False}):

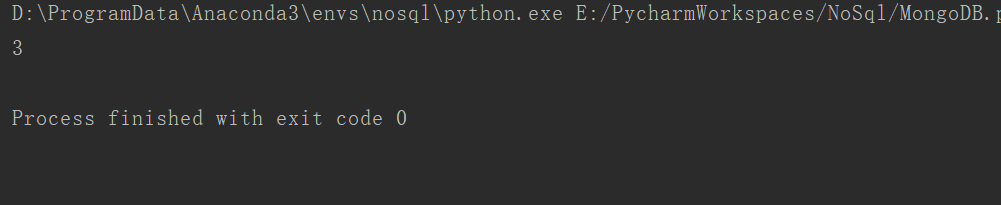
print(r)



## 聚合查询

print(col.find({"name":"fruits"}).count())

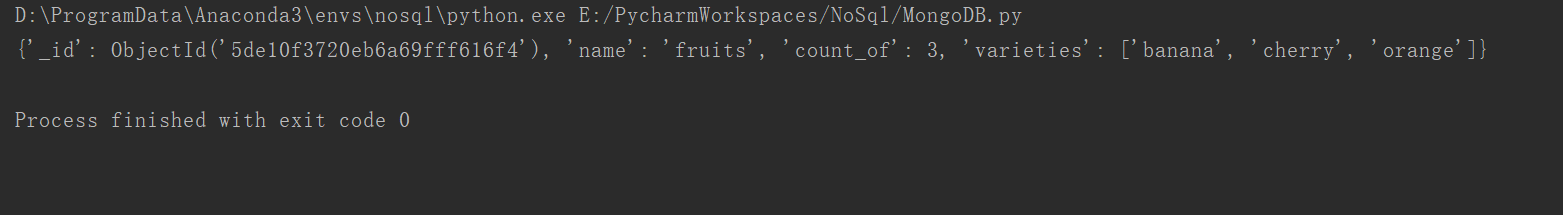
print(col.count\_documents({"name":"fruits"}))



## 比较运算符

for r in col.find({"count\_of":{"$lt":4}}):

print(r)



## 地理索引查询

from pymongo import GEO2D

db.places.create\_index([("loc",GEO2D)])

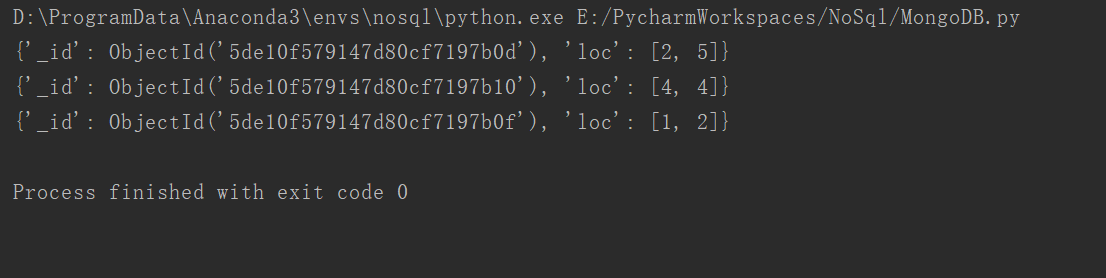
#插入经纬度信息

result = db.places.insert\_many([{"loc": [2, 5]},{"loc":[30, 5]},{"loc": [1, 2]},{"loc": [4, 4]}])

# $near 查询

for doc in db.places.find({"loc": {"$near": [3, 6]}}).limit(3):

print(doc)



## Gridfs操作

import gridfs

fs=gridfs.GridFS(db)

fileid=fs.put(b"hello word",filename="testfile")

print(fs.exists({'filename':'testfile'}))

print(fs.list())

for doc in fs.find({"filename":"testfile"}):

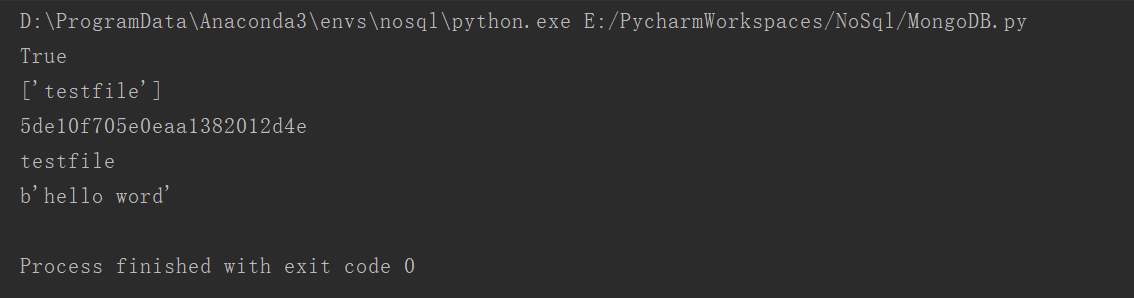
print(doc.\_id)

print(doc.filename)

print(doc.read())

#删除

fs.delete(doc.\_id)



## 删除数据

col.delete\_one({"name":"fruits"})

print(col.find\_one())

## 删除集合

db.places.delete\_many({})

db.drop\_collection("testcol")