1.非关系型数据库

database(数据库)-->collection(集合)>document(文档)>JSON

操作非关系型数据库,用js语言操作MongoDB数据库

2.安装 不能选择安装MongoDB compass ,安装完成后需要配置数据的存储目录(mongod --dbpath="位置"),否则启动不成功.

mongo默认连接 本地的test数据库,远程连接 mongo ip:port

默认的端口27017

3.use 名称 创建数据库(存在的时候切换数据库) use mymdb 创建了一个mymdb数据库

db.dropDatabase() 删除数据库(注意大小写)

show dbs 查看所有数据库

db 查看当前所在数据库

show collections 或 show tables 查看当前数据库的所有集合

创建集合:分为显示创建,和隐式创建

db.createCollection("集合名称") 显示创建集合

db.集合名称.insert({name:"jack"}) 注意添加的key是否有引号(没有引号仍然可以插入),

db.名称.insert({内容}) 向集合中添加document文档,

并且支持循环语句输入内容文档

db.connection.remove({筛选条件}) 删除掉符合筛选条件的文档,

db.connection.find(条件) 筛选符合条件的(不输入查询条件,查询所有)

db.connection.findOne() 查询第一个文档

mongoDB 的用户名密码(登录方式)

2018-12-17

查询集合中的文档,返回特定的键值

1.返回除了某个字段的其他字段

db.collection.find({},{"字段":0}) 0表示不返回,1表示只返回

2.MongoDB的条件表达式:

db.collection.find({"字段":{条件}})

大于 $gt:value ,小于$lt:value,大于等于 $gte:value

小于等于 $lte :value 不等于 $ne:value

3.常用操作

统计总个数: db.collection.count();

统计满足条件的个数db.collection.find({}).count()

排序:db.collection.find().sort({"字段":"1"}) 1表示正向排序,-1 表示逆向排序

分页:db.collection.find(.)skip(开始的索引).limit(每页的条数)

db.collection().find().skip(开始的索引).limit(每夜的条数).count(0表示计算的总条数,其他的表示当前页可以存放多少条,意思是没有数据仍然会返回每页的条数)

$all主要用来查询数组中的包含关系

db.collection.find({"数组字段":{$all:[其中的数据不是like,必须完全一致]}})

$in 相当于in

db.collection.find({age:{$in:[12,123]}})

$nin 与$in相反

$or或者

db.collection.find({$or:[{条件1},{条件2}]})

$nor 过滤掉不是那几种条件的数据,

$exists 查询某个可以存在的文档

db.collection.find({"name":{$exists:1}}) 1表示存在,0表示不存在

4.查询集合中的文档,含有类似于有表达概念

var x=db.collection.find();

x.hasNext();

x.next();

5.更新文档的操作

db.collection.update(criteria,objNew,upsert,multi)

criteria,用于设置条件查询的对象,

objNew用于设置更新内容的对象,

upsert:如果科技路已经存在,更新,否则新增,取值0或1

multi :如果存在多个符合条件的记录,是否全部更新,取值为0或者1

默认情况下,只会更新第一条数据

一般情况,后两个参数为0,1

存在问题,需要更新的值必须保存完整,否则,会失去原有的数据,

更新某个数据值加1,其他键值不发生变化,

$inc

db.collection.update({"name":1},{$Inc:{"字段":增减值}})

$set 设置某个键值,如果键值不存在则创建

db.collection.update({"name":"jack"},{$set:{"age":22}})

$unset 用来删除某个键值

db.collection.update({"name":"jack"},{$unset:{"age":22}})

6.索引

创建普通索引,

db.collection.ensureIndex({key:1})

相关信息:db.collection.stats()

查看索引的情况,db.collection.find().explain()

删除索引 db.collection.dropIndex({key:1})

删除集合,会删除全部索引

创建唯一索引

db.collection.ensureIndex({key:1},{unique:true})

7.固定集合

指的是显示创建并且大小固定的集合.

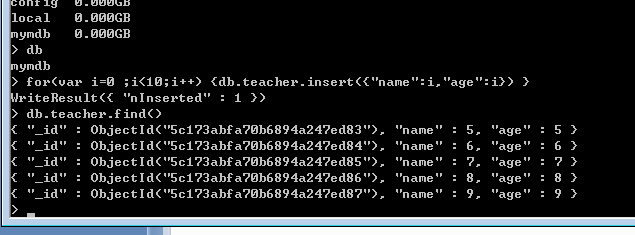
特点:固定集合很像环形队列,如果空间不足,最早的文档就会被删除,为新的文档腾出空间,一般来说,固定集合适用于任何想要自动淘汰过期属性的场景,没有过多的操作限制.

创建一个固定集合:

db.createCollection("集合名称",{capped:true,size:100000,max:100})

size指定集合的大小,单位kb,max指定文档的数量 capped 表示固定的意思

当指定文档数量时,必须同时知道大小.淘汰机制只有在容量还没有满时会根据文档数量来工作,要是容量满了,淘汰机制会依据容量来工作



将普通集合转换为固定集合

db.runCommand({convertToCapped:"集合名称",size:100000,max:100})

普通集合转换为固定集合如果原先的文档数量已经大于max,淘汰机制会依据size来工作.

8.备份(Mongodump)和恢复(mongorestore)

备份数据命令:

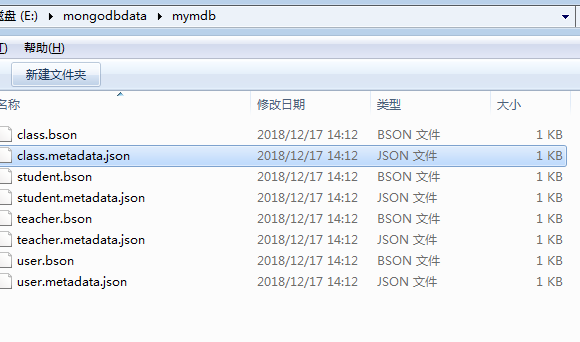
mongodump -h dbhost -d dbname -o dbdirectoty

-h服务器地址

-d 需要备份的数据库实例

-o 备份数据存放位置,目录需要提前建立

mongodump -h 127.0.0.1:27017 -d mymdb -o e:\mongodbdata



数据格式为bson(二进制数据)和json

恢复数据,

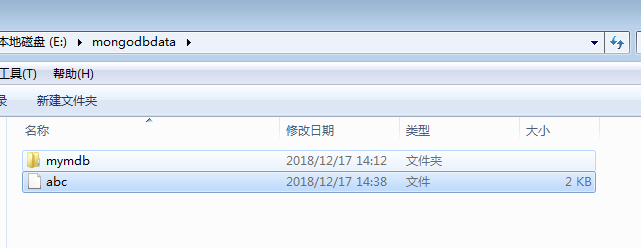
mongorestore -h dbhost -d dbname -directoryperdb dbdirectory

红色字体不需要存在 ,会影响选择的数据库名称

9.导入(mongoimport)和导出(mongoexport)

导出数据:mongoexport -h dbhost -d dbname -c collectionName -o output

-c 要导出的集合的 -o 导出的文件名



导入数据

mongoimport -h -d -c output (出现含有objectid无法导入的问题)

10.安全和认证

每个数据库中可以有多个用户,如果开启了安全性检查,只有数据库认证用户才能执行读或者写操作,在认证的上下文中,MongoDB会将普通的数据作为admin数据库处理,admin数据库中的用户会被视为超级用户(管理员).在认证之后,管理员可以读写所有数据库,执行特定的管理命令,如listDatabases和shutdown

在开启安全检查之前,一定要有一个管理员账号

在admin数据库中创建管理员账号

use admin

db.addUser("root","root")

在普通数据库中,创建用户

db.addUser("张三","11") 拥有读写权限

db.addUser("lis","123",true) 只有读权限

重启数据库服务,并且开启安全检查

mongod高版本已经不支持addUser

支持的格式是

db.createUser(user,writeConcern)

db.createUser(

user:"name",

pwd:"word",

customData:{为任意内容,}

roles:[赋给用户的角色如果是空数组则不设定角色]

)

11.主从复制

主从复制是MongoDB最常用的复制方式,可用于备份,故障恢复,读扩展.

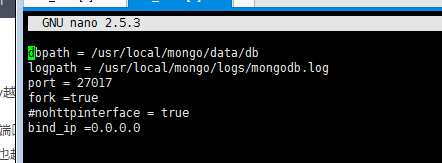
最基本的设置方式,就是建立一个主节点和一个或者多个从节点,

启动主节点:

mongod --dbpath d:\master --port 10000 --master

启动从节点:

mongod --dbpath d:\salve --port 10001 --slave --source localhost:10000(没有实现)



自己创建的配置文件

启动时需要./mongod -f mongodb.conf

重启:bind\_ip可以解决ip无法访问的问题

创建主节点的配置

logappend: true 日志的填写默认为覆盖,设为true为追加

journal: 日志

fork : 用户认证,没有用户也可以操作,直到创建了用户,

12.副本集 就是有自动故障恢复功能的主从集群

主从集群和副本集最大的区别就是副本集没有固定的主节点,个集群会选出一个主节点,当其挂掉后,又在剩下的从节点中选取其他从节点为主节点,,副本集总有一个活跃点和一个或者多个备份节点

13.存储mongo的数据库位置必须自己创建,启动程序是不会自己创建的

14.java操作mongodb的增删改查



15.总结其他的$命令

db.collection.命令({"字段":{条件}})

$set 更新数据保持原有不需要修改的键值

大于 $gt:value ,小于$lt:value,大于等于 $gte:value

小于等于 $lte :value 不等于 $ne:value

$all主要用来查询数组中的包含关系,其中的数据不是like,必须完全一致

$in 相当于in

$nin 与$in相反

$or或者

$nor 过滤掉不是那几种条件的数据,

$exists 查询某个可以存在的文;1表示存在,0表示不存在

$inc 更新某个数据值加1,其他键值不发生变化,

$set 设置某个键值,如果键值不存在则创建

$unset 用来删除某个键值