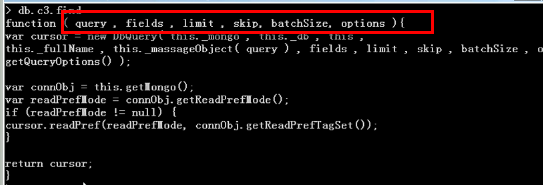
说明：下面实例中的c1或者c2集合，其实可以理解为就是数据库中的表

Find方法需要理解各个参数



连接客户端



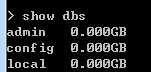
# 常规命令

## 注销用户



## show dbs

展示所有库（这里展示的是在硬盘上的）



## db

展示当前所使用的库（如果是内存库，上面的命令是看不到现在使用的库的）



## 切库/新建数据库

**use [dbname]**

如果dbname还没创建则会自动新建，否则直接切库



## 删除数据库（只能删除当前数据库）



## 隐式创建数据

在创建集合的同事王集合里面添加数据

在数据库中创建了一个c1的集合对象



## 显示创建

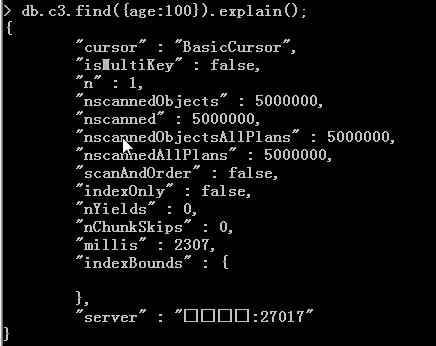
只是创建了一个c2集合对象，没有数据



## 显示数据库中所有对象



## 分析查询过程



## 查找所有数据

查找c1数据，objectId是mongodb自己维护的（自动生成的）



## 继续往c1添加数据



## 根据条件查询数据

其实这里的c1就是表，name就是表里面的字段



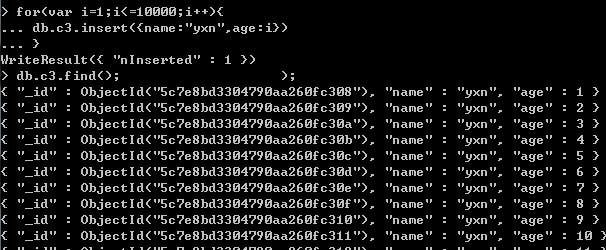
## 只返回所需要的字段

这里就只会返回age



## 写js代码

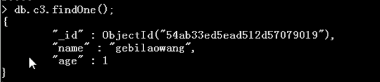
虽然插入了很多条，但是mongo默认只显示前20条



## 查看数据条数



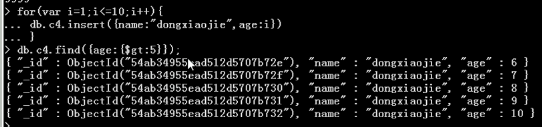
## 查找集合中的第一条记录（也就是查表中top1的意思）



## 根据条件删除数据



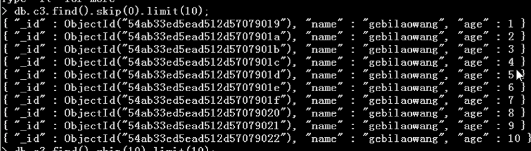
## 查看大于/大于等于/小于/小于等于/不等于某个值的数据



**gt大于 gte大于等于 lt小于 lte小于等于 ne不等于**

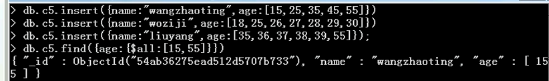
## 分页

查询1到10条

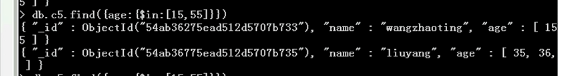


## $all查询满足所有条件的数据

$all后的数据有一个不满足就不会返回

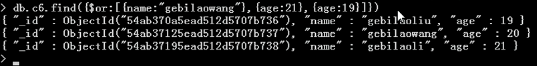


## $in满足其中一条即可取出



$nin与in相反

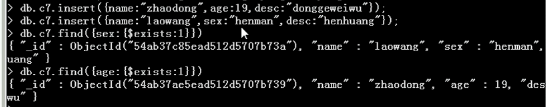
## 或者查询（nor为相反）



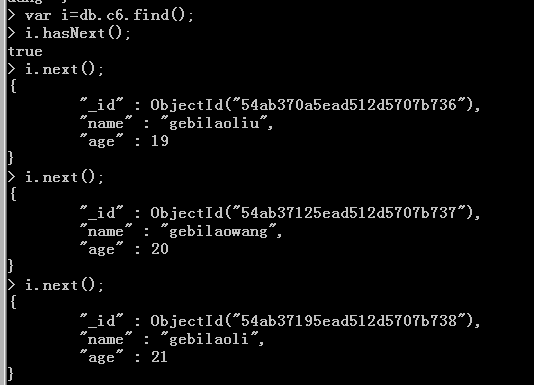
## 查询是否包含某个键（字段）的数据

Exists:1表示包含

Exists:0 表示不包含

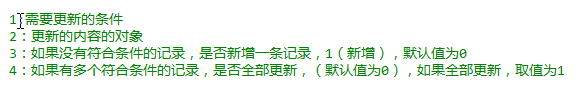


## 游标的使用



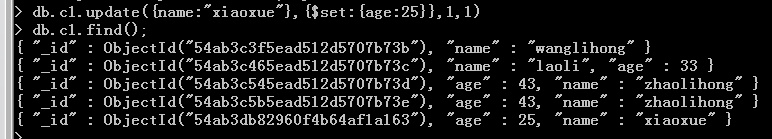
## 更新

一共四个参数：

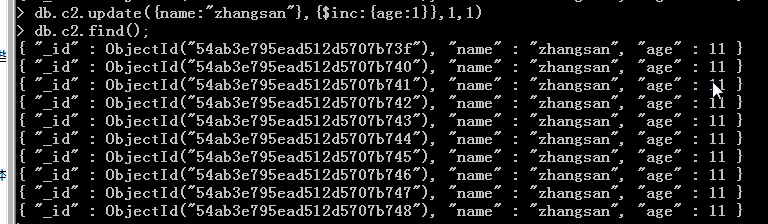


第一个参数如果是{}表示更新所有

第四个参数为0的时候，执行更新只会更新满足条件的其中一条数据。



## 更新（对数值数据进行加减）



## 更新（删除某一个key）$unset



# 索引（可以根据常用命令7.来分析索引是否建立成功）

## 查询数据库索引信息



## 对字段建立索引（普通索引）



## 删除索引



## 建立唯一索引

唯一索引绑定的key，value是不允许有相同的



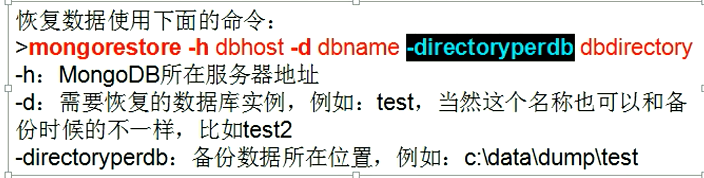
# 数据库备份/还原

## 备份

**Mnongodump –h 地址:端口号 –d 所要备份的数据库名 -o 所要备份的地址**



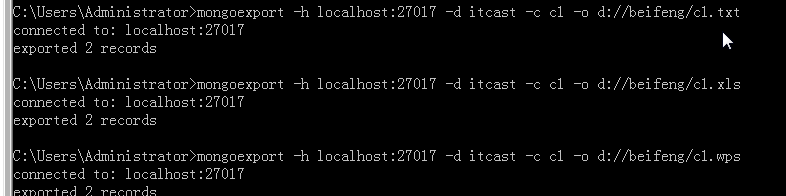
## 恢复数据库



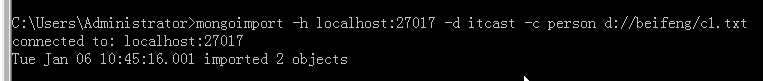


## 数据的导入导出

大多是导出集合数据，相当于导出表数据



导入



# 固定集合

capped：表示是否固定

size：表示保存数据的总大小单位KB

max：表示可以保存的文档最大个数

**若当size或者max到达设定的值，程序不会报错，而会把最初插入的数据删除。**

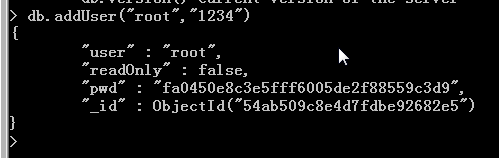


# 数据库安全

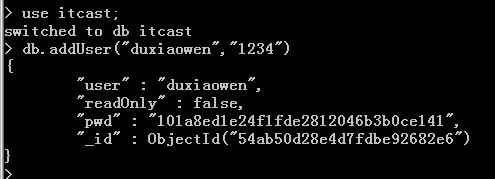
在开启安全检查之前，需要创建一个管理员账号，admin数据库中的用户会被视为超级管理员。

## 进入admin数据库中，添加用户（这里面的用户会被视为超级管理员）

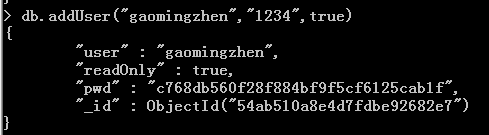




## 有了超级管理员后就可以为其他数据库分配用户



下图创建的用户只读



## 服务器开启安全检查（这步是在服务端操作的）





## 客户端连接，连接时需要认证

# Mongodb集群（主从复制，读写分离）

先在磁盘上创建两个文件加，一个主数据库的数据保存目录，一个从数据库的数据保存目录。

声明主节点：--master



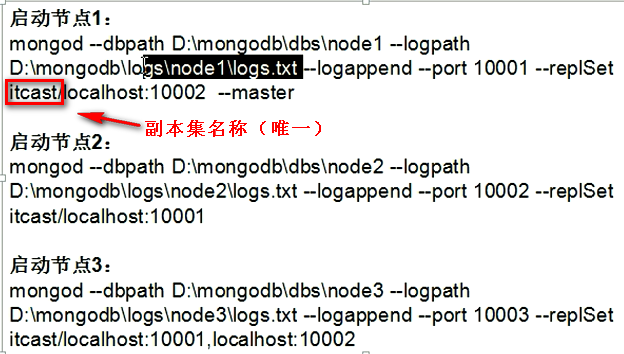
声明从节点（从节点默认不能插入数据，只能读取数据）



# 副本集（和集群的区别在于副本集有自动故障恢复的功能）

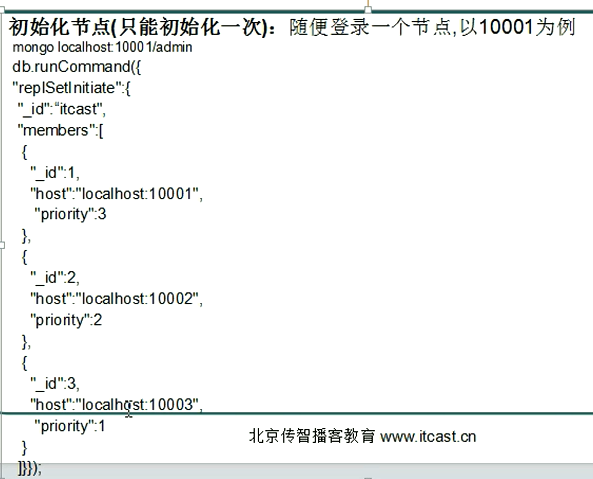
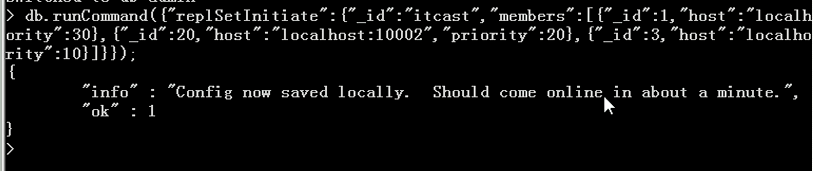
也就是说副本集中的主节点挂掉后，会自动选取一个从节点作为主节点。而且从节点平常并没有什么用（不能查询也不能插入数据），只有当从节点转变为主节点他才有用处。



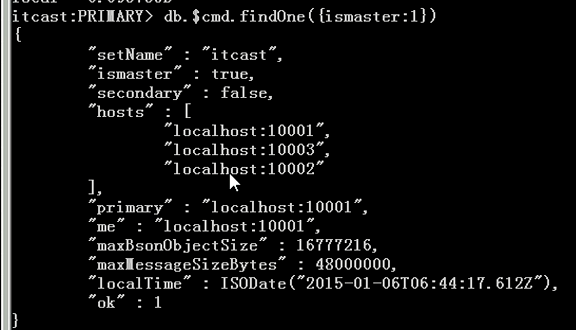




Priority越大优先级越高

查询副本集信息



# 数据分片

## 先创建分片数据存放的目录，还要创建一个cofig目录（用来记录mongodb路由规则，数据与分片的关系）

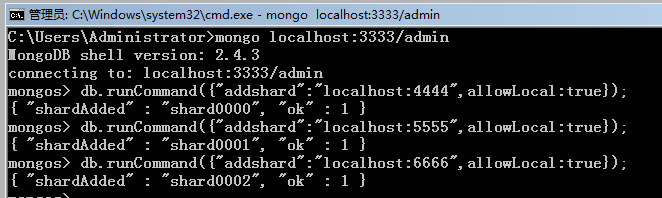
启动这些服务器

## 启动mongos进程（路由器）



## 将路由与分片服务端建立连接

AllowLocal:为true，表示客户端可以不通过路由器直接连接分片服务端



## 启用指定数据库的分片功能



## 指定分片的集合与分片键

