MongoDB

# MongoDB简介

MongoDB 是由C++语言编写的，是一个基于分布式文件存储的开源数据库系统。在高负载的情况下，添加更多的节点，可以保证服务器性能。MongoDB 旨在为WEB应用提供可扩展的高性能数据存储解决方案。MongoDB 将数据存储为一个文档，数据结构由键值(key=>value)对组成。MongoDB 文档类似于 JSON 对象。字段值可以包含其他文档，数组及文档数组。

# MongoDB基本概念



MongoDB可以使用的类型

# MongoDB基本操作

## 创建数据库目录

MongoDB的数据存储在data目录的db目录下，但是这个目录在安装过程不会自动创建，所以你需要手动创建data目录，并在data目录中创建db目录

## 命令行中运行 MongoDB 服务

在安装目录的bin目录下执行 ./mongod –dbpath xxxx

如果执行成功，就会处于等待连接状态，等待连接

## 基本命令使用

### 一般命令

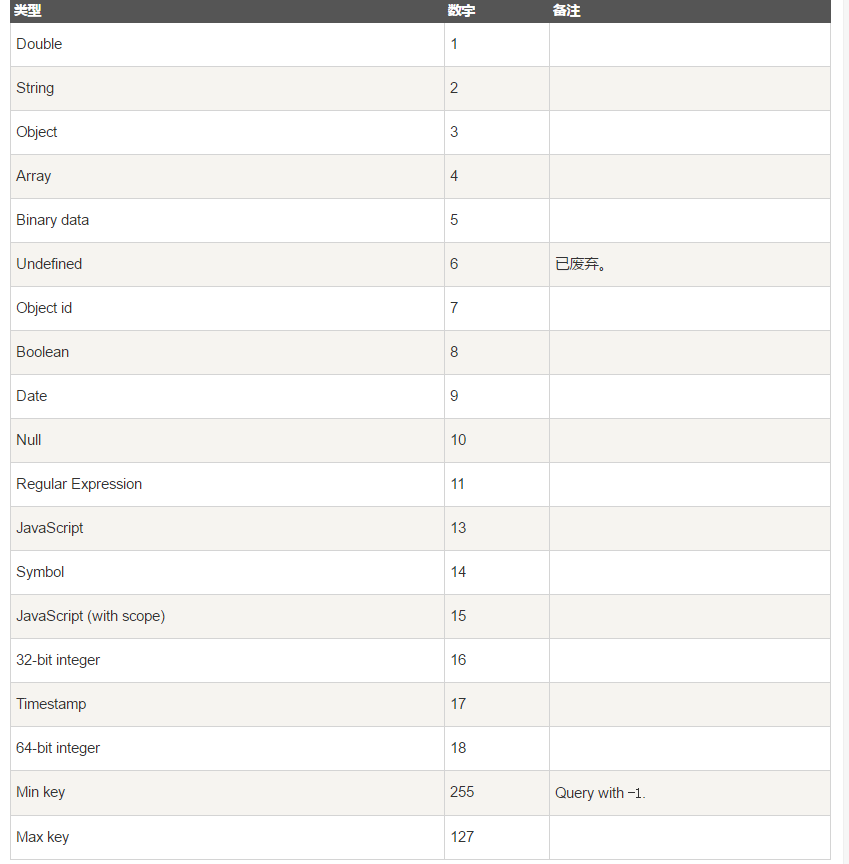
创建数据库 use DATABASE\_NAME

删除数据库 db.dropDatabase()

删除集合 db.collection.drop()

插入文档 db.COLLECTION\_NAME.insert(document)

MongoDB可以使用的类型：



如果想获取 "col" 集合中 title 为 String 的数据，你可以使用以下命令：

db.col.find({"title" : {$type : 2}})

## Map-reduce操作

Map-Reduce是一种计算模型，简单的说就是将大批量的工作（数据）分解（MAP）执行，然后再将结果合并成最终结果（REDUCE）

db.collection.mapReduce(

function() {emit(key,value);}, //map 函数

function(key,values) {return reduceFunction}, //reduce 函数

{

out: collection,

query: document,

sort: document,

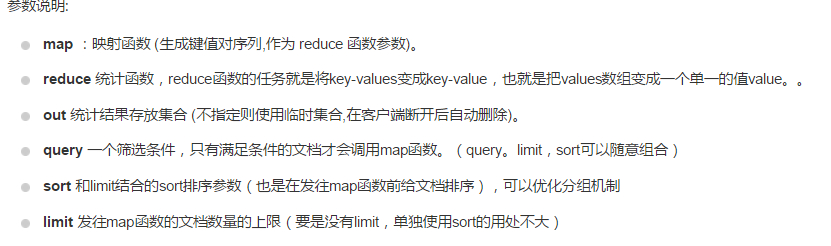
limit: number

}

)

使用 MapReduce 要实现两个函数 Map 函数和 Reduce 函数,Map 函数调用 emit(key, value), 遍历 collection 中所有的记录, 将 key 与 value 传递给 Reduce 函数进行处理。

Map 函数必须调用 emit(key, value) 返回键值对。



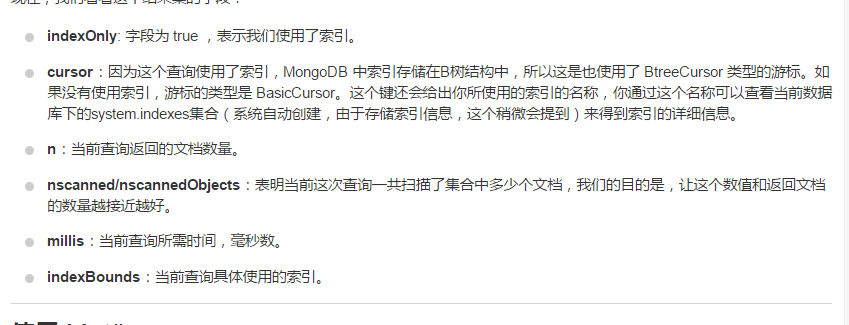
### MongoDB查询分析语句

MongoDB 查询分析可以确保我们建议的索引是否有效，是查询语句性能分析的重要工具。

MongoDB 查询分析常用函数有：explain() 和 hint()

db.users.find({gender:"M"},{user\_name:1,\_id:0}).explain()

返回结果集中的主要字段介绍：



 hint 来强制 MongoDB 使用一个指定的索引。

其他命令如条件查找，排序，索引，聚合等可以自行百度

# MongoDB Java

连接到MongoDB数据库

import java.util.ArrayList;

import java.util.List;

import com.mongodb.MongoClient;

import com.mongodb.MongoCredential;

import com.mongodb.ServerAddress;

import com.mongodb.client.MongoDatabase;

public class MongoDBJDBC {

public static void main(String[] args){

try {

//连接到MongoDB服务 如果是远程连接可以替换“localhost”为服务器所在IP地址

//ServerAddress()两个参数分别为 服务器地址 和 端口

ServerAddress serverAddress = new ServerAddress("localhost",27017);

List<ServerAddress> addrs = new ArrayList<ServerAddress>();

addrs.add(serverAddress);

//MongoCredential.createScramSha1Credential()三个参数分别为 用户名 数据库名称 密码

MongoCredential credential = MongoCredential.createScramSha1Credential("username", "databaseName", "password".toCharArray());

List<MongoCredential> credentials = new ArrayList<MongoCredential>();

credentials.add(credential);

//通过连接认证获取MongoDB连接

MongoClient mongoClient = new MongoClient(addrs,credentials);

//连接到数据库

MongoDatabase mongoDatabase = mongoClient.getDatabase("databaseName");

System.out.println("Connect to database successfully");

} catch (Exception e) {

System.err.println( e.getClass().getName() + ": " + e.getMessage() );

}

}

}

其他基本操作：

使用com.mongodb.client.MongoDatabase 类中的createCollection()来创建集合

使用com.mongodb.client.MongoDatabase类的 getCollection() 方法来获取一个集合

使用com.mongodb.client.MongoCollection类的 insertMany() 方法来插入一个文档

使用 com.mongodb.client.MongoCollection 类中的 find() 方法来获取集合中的所有文档。

使用 com.mongodb.client.MongoCollection 类中的 updateMany() 方法来更新集合中的文档。

具体其他操作详见API

# MongoDB集群搭建

参看了一篇博客地址：http://blog.csdn.net/luonanqin/article/details/8497860

# MongoDB注意事项

## 1 mongodb不支持事务

mongodb不支持事务，所以，在你的项目中应用时，要注意这点。无论什么设计，都不要要求mongodb保证数据的完整性。

但是mongodb提供了许多原子操作，比如文档的保存，修改，删除等，都是原子操作。

## 2索引

索引不能被以下的查询使用：

正则表达式及非操作符，如 $nin, $not, 等。

算术运算符，如 $mod, 等。

$where 子句

所以，检测你的语句是否使用索引是一个好的习惯，可以用explain来查看。

## 3.ObjectId

ObjectId 是一个12字节 BSON 类型数据，有以下格式：

前4个字节表示时间戳

接下来的3个字节是机器标识码

紧接的两个字节由进程id组成（PID）

最后三个字节是随机数。

MongoDB中存储的文档必须有一个"\_id"键。这个键的值可以是任何类型的，默认是个ObjectId对象。

在一个集合里面，每个文档都有唯一的"\_id"值，来确保集合里面每个文档都能被唯一标识。

MongoDB采用ObjectId，而不是其他比较常规的做法（比如自动增加的主键）的主要原因，因为在多个 服务器上同步自动增加主键值既费力还费时