

第三章 文档数据库

串讲归纳

3.0 前言

在上一节中我们提到过：非关系型数据库，也就是NoSQL——强调Key-Value存储非关联的文档数据库。

为什么需要这样的数据库呢？因为传统的关系型数据库已经无法满足快速查询与插入数据的需求，它通过降低数据的安全性，减少对事务的支持，减少对复杂查询的支持，获取性能上的提升。

3.1 文档数据库MongoDB

这是其中一款较为热门的文档数据库，基于分布式进行高性能文件存储。

3.1.1 特点

总的来说：高可用、高水平扩展

1. 数据文件**存储格式为BSON**（Binary JSON），JSON（JavaScript Object Notation）的扩展。像这样：

```
{
  name: "sue",
  age: 26,
  status: "A",
  groups: [ "news", "sports" ]
}
```

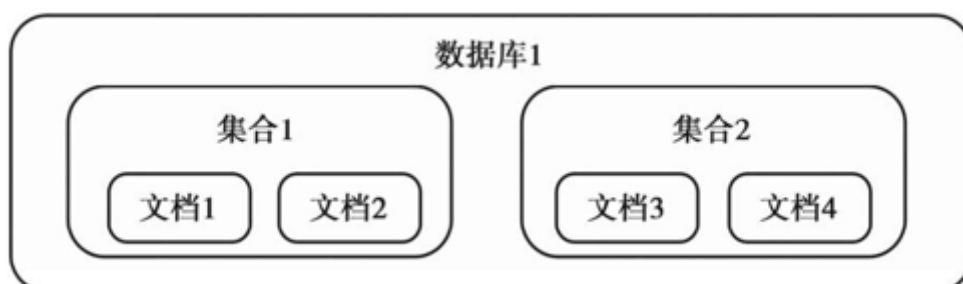
← field: value
← field: value
← field: value
← field: value

2. **面向集合存储，简单、快速且易于存储。**

📌 Important

一个文档是文档型数据库的一条记录，就像上面的图片：存你的一套个人信息【name:zhangsan;age:20等等】，这也称为元数据（Meta），你这个对象就是元，你相关的数据就是元数据，一个文档通常存一个这样的元数据。

一个集合就是一组文档，一个数据库就是一组集合。关系如下：



要注意，集合有这样的特点：无模式，也就是对文档的字段无要求，可以是你的个人信息，也可以是母猪的产后护理。但是如果需要，比如存储你们班级的个人信息，也可以适用格式校验功能进行字段一致性管理。

3. 模式自由，具有良好的水平可扩展性。已经说过。

4. 支持动态查询

可以很方便地查询内嵌文档和对象数组，几乎可以实现类似关系型数据库单表查询的绝大部分功能。

5. 完整的索引支持

文档内嵌对象和数组都可以创建索引。

6. 二进制数据存储

MongoDB使用传统高效的二进制数据存储方式，可以将图片文件甚至视频转换成二进制的数据存储到数据库中。

其余的不赘述，没必要记也记不住。

3.1.2 数据类型

JSON只有6中数据类型，null，布尔，数字，字符串，数组和对象。MongoDB又新增了些。

序号	类型	名称	描述
1	ObjectId	唯一标识文档的特殊数据类型	文档自动生成的_id，类似于唯一主键，可以很快地生成和排序，它包含24B。0~8B (5d69cf29) 表示时间戳，即这条数据产生的时间。9~14B (669387) 表示机器标识符，即存储这条数据时的机器编码。15~18B (1a67) 表示进程id。19~24B (f19983) 表示计数器。
2	String	字符串	UTF-8字符串都可表示为字符串类型的数据。
3	Boolean	布尔值	只有两个值true和false（注意true和false首字母小写）。
4	NumberInt/ NumberLong	整数	一般有32位对于整数，可以使用 NumberInt 或 NumberLong 类，它们分别表示 4 字节和 8 字节的有符号整数。
5	Double	浮点数	MongoDB中没有Float类型，所有小数都是Double类型
6	Arrays	数组	值的集合或者列表可以表示成数组，数组中的元素可以是不同类型的数据
7	Date	日期	MongoDB 会将日期存储为 64 位整数，表示自 Unix 纪元（1970 年 1 月 1 日）以来的毫秒数，不包含时区信息
8	Null	空	空值或不存在的字段

3.3.3 规范化

文档数据库未规范化，不同的文档间可能重复，也可能有差异。比如你今天20岁，文档A记录了age:20，明天21岁，就会有新的文档B记录age:21。

3.2 MongoDB的使用

⚠ Warning

这部分会重点考察，但千万不要死记，结合后面题目一起学习记忆，这里先过一遍有个印象。

3.2.1 数据库常用操作

序号	命令	描述
1	show dbs	查看所有数据库
2	db 或 db.getName()	查看当前使用的数据库
3	db.stats()	显示当前db状态
4	db.version()	查看当前db版本
5	db.getMongo()	查看当前db的连接服务器机器地址
6	db.dropDatabase()	删除当前使用数据库
7	db.getPrevError() db.resetError()	查询之前的错误信息和清除
8	use db	创建

数据库命名规范：基本上，只能使用ASCII中的字母和数字，符号禁用；区分大小写，建议全部小写；不能是空字符串。

3.2.2 集合常用操作

查看数据库中集合：使用 show collections 或 show tables

创建：db.createCollection(name, options)

删除：db.collection.drop()

⚠ Caution

在 MongoDB 中，集合只有在内容插入后才会创建！就是说，创建集合（数据表）后要再插入一个文档（记录），集合才会真正创建。

命名规范：不能包含\0字符（空字符）；不能以“system.”开头，这是为系统集合保留的前缀；不能在集合名中包含保留字符\$。

3.2.3 文档常用操作

简称CRUD：create ---insert； read --find； update； delete

1.写入

db.COLLECTION_NAME.insert(document)

其中document：满足BSON格式的一个或多个文档，多个文档间用逗号分割。也有专用的insertOne。

💡 Tip

每次插入文档时，集合中会为此文档生成一个名为“id”的默认主键字段。这个主键名称是固定的，可以是MongoDB支持的任何数据类型，插入数据时不指定“id”字段值时，则默认值是ObjectId。

2.更新

```
db.collection.update( , , { upsert: , multi: , writeConcern: } )
```

参数说明【后面做题来学习，不要死记】：

•**query**：update的查询条件。•**update**：update的对象和一些更新的操作符（如\$，\$inc...）等，•**upsert**：可选，这个参数的意思是，如果不存在update的记录，是否插入objNew，true为插入，默认是false，不插入。•**multi**：可选，mongodb默认是false,只更新找到的第一条记录，如果这个参数为true,就把按条件查出来多条记录全部更新。•**writeConcern**：可选，抛出异常的级别。

3.查询

```
db.collection.find(query, projection)
```

查询方式	举例	说明
与操作	db.user.find({ "name": "mongo" , "age": 20 }) ;	查询同时满足 name 字段为 mongo, age 字段为 20 的文档
或操作 \$or	db.user.find({ \$or: [{ "age": 23 } , { "age": 20 }] }) ;	查询满足 age 字段值为 23 或 20 的文档
大于 \$gt	db.user.find({ "age": { \$gt: 20 } }) ;	查询 age 字段值大于 20 的文档
小于 \$lt	db.user.find({ "age": { \$lt: 20 } }) ;	查询 age 字段值小于 20 的文档
大于等于 \$gte	db.user.find({ "age": { \$gte: 20 } }) ;	查询 age 字段值大于等于 20 的文档
小于等于 \$lte	db.user.find({ "age": { \$lte: 20 } }) ;	查询 age 字段值小于等于 20 的文档
是否存在 \$exists	db.user.find({ "age": { \$exists: true } }) ;	查询 age 字段存在的文档
不等于 \$ne	db.user.find({ "age": { \$ne: 20 } }) ;	查询 age 字段值不等于 20 的文档
包含 \$in	db.user.find({ "name": { \$in: ["mongo" , "mary"] } }) ;	查询 name 字段值在数组 ["mongo" , "mary"] 中的文档
不包含 \$nin	db.user.find({ "name": { \$nin: ["mongo" , "mary"] } }) ;	查询 name 字段值不在数组 ["mongo" , "mary"] 中的文档
取反 \$not	db.user.find({ \$not: { "name": "mongo" , "age": 20 } }) ;	查询查询同时不满足 name 字段为 mongo, age 字段为 20 的文档。所有的查 询条件都可以进行取反操作

4.删除

```
db.collection.remove(, {justOne:,writeConcern: document> } )
```

3.2.4 安全认证

创建管理员 db.createUser({user:"fox",pwd:"fox",roles:["root"]})

查看所有用户信息 show users

删除用户 db.dropUser("fox")

创建应用数据库用户 use appdb, db.createUser({user:"appdb",pwd:"fox",roles:["dbOwner"]})

3.3 游标

3.3.1 介绍

1. 游标定义

游标是用于查询结果集的指针。当执行查询时，MongoDB会返回一个游标，然后可以使用游标来逐个获取查询结果中的文档。游标可以用于遍历大量数据，支持分页查询，以及在查询过程中进行一些操作。

2. 游标的类型

- (1) 客户端游标：由客户端创建和管理，适用于单次查询操作。
- (2) 数据库游标：由数据库服务器创建和管理，适用于跨多个文档的查询操作。

3.3.2 使用

MongoDB 游标提供了一系列方法，用于操作游标中的数据。

- (1) `find()`: 可以创建一个游标
- (2) `next()`: 获取游标中的下一个文档。
- (3) `toArray()`: 将游标中的所有文档返回为一个数组。
- (4) `forEach()`: 遍历游标中的所有文档。
- (5) `count()`: 返回游标中的文档数量。
- (6) **`limit()`: 限制游标返回的文档数量。**

💡 Tip

限制结果数量，可以在 `find` 之后链式调用 `limit` 函数，语法格式：

```
db.COLLECTION_NAME.find().limit(options)
```

参数说明：options：可选项，数字，用于指定从MongoDB中读取的记录条数，如果没有指定`limit()`方法中的参数则显示集合中的所有数据。

- (7) **`skip()`: 跳过游标中的前n个文档。**

💡 Tip

跳过指定数量的数据，语法格式：

```
db.COLLECTION_NAME.find().limit(NUMBER).skip(NUMBER)
```

参数说明：NUMBER：数字，作为跳过的记录条数。

- (8) **`sort()`: 对游标中的文档进行排序。**

💡 Tip

对数据进行排序，语法格式：

```
db.COLLECTION_NAME.find().sort({KEY:NUMBER})
```

参数说明：KEY：指定排序的字段；NUMBER：1或-1，指定排序方式，其中1为升序排列，-1为降序排列。

MongoDB 的类型排序规则，从最小到最大：最小值null；数字（整型、长整型、双精度浮点型、小数型）；字符串；对象/文档；数组；二进制数据；对象 ID；布尔型；日期；时间戳

(9) Limit、skip和sort结合使用实现分页

💡 Tip

假设我们运营一个在线商店，有人想搜索mp3。若我们希望将查询结果按照价格从高到低排序，并每页显示 50 个结果。可以通过以下查询实现：

```
db.stock.find({"desc": "mp3"}).limit(50).sort({"price": -1})
```

下一页：db.stock.find({"desc": "mp3"}).limit(50).skip(50).sort({"price": -1})

3.3.3 特点

1.优缺点

优点：简单易用，适合小规模的数据分页。

缺点：如果数据量特别大，skip 的性能会下降，因为需要先找到被略过的结果，然后再丢弃这些数据。

2.改进

不使用skip对结果进行分页来提高效率：基于最后一个文档的字段”的方式来实现分页；在文档中添加随机键

3.生命周期

边用边放：尽快释放游标。在服务器端，游标会占用内存和资源。一旦游标遍历完结果之后，或者客户端发送一条消息要求终止，数据库就可以释放它正在使用的资源。

特殊情况：（1）遍历完（2）游标超出客户端的作用域时（3）10 分钟没有被使用

3.4 索引

3.4.1 介绍

1.索引定义

索引类似书的目录，极大地提高了查找速度；而**不使用索引**的查询称为**集合扫描**，就是遍历整个集合来查询。

💡 Tip

索引本质上是树，最小的值在最左边的叶子上，最大的值在最右边的叶子上，使用索引可以提高查询速度（而不用全表扫描）

2.索引分类

索引类型	用途
单字段索引 (Single Field Index)	针对某一键 key 创建了单字段索引，其能加速对 key 字段的各种查询请求，是最常见的索引形式，MongoDB 默认创建的 id 索引也是这种类型。
复合索引 (Compound Index)	复合索引是单字索引的升级版，它针对多个字段联合创建索引，先按第一个字段排序，第一个字段相同的文档按第二个字段排序，依次类推。

索引类型	用途
多 key 索引 (Multikey Index)	当索引的字段为数组时，创建出的索引称为多 key 索引。
哈希索引 (Hashed Index)	按照某个字段的hash值来建立索引，目前主要用于 MongoDB Sharded Cluster 的 Hash 分片，哈希索引只能满足字段完全匹配的查询，不能满足范围查询等。
地理位置索引 (Geospatial Index)	能很好的解决 O2O 的应用场景，比如：查找附近的美食、查找某个区域内的车站等。
文本索引 (Text Index)	能解决快速文本查找的需求，比如有一个博客文章集合，需要根据博客的内容来快速查找，则可以针对博客内容建立文本索引。

3.索引使用

(1) 创建索引

MongoDB使用createIndex () 方法来创建索引，语法格式：`db.collection.createIndex(keys, options)` 其中，Keys：要创建的索引字段和索引顺序（1或-1，1为指定按升序创建索引，-1为指定按降序创建索引）构成的键值对。

💡 Tip

优缺点：索引可以显著缩短查询时间，然而，使用索引是有代价的：修改索引字段的写操作（插入、更新和删除）会花费更长的时间。而且要合理选择索引字段（如你应选择学号作为索引而不是性别）。

(2) 复合索引

索引只能当作“目录”来查阅吗？不！还可以进行高效的排序，比如你一定不希望1班和2班的学生信息混杂在一起。然而，索引在用作排序前缀时才能发挥作用。所以，应该先按class排，再按学号ID排。

(3) 查询类型和索引方式

等值查询【学号ID=20220101】

💡 Tip

举例：`db.users.find({"age": 21}).sort({"ID": -1})`

由于可能有多个文档具有该值。多亏了索引中的第二个字段 ID，MongoDB 可以确保返回的结果已经按 ID降序排序。

优势：高效；无需排序；灵活

范围查询【18<=年龄Age<=21】

💡 Tip

举例：`db.users.find({"age": {"$gte": 21, "$lte": 30}})`

优势：查询速度快、资源利用率低、结果按顺序返回，并且分页处理更加高效。

多值查询【18<=年龄Age<=21且BMI==正常】

💡 Tip

举例：`db.users.find({"age": {"$gte": 21, "$lte": 30}}).sort({"username": 1})`

效率较低；受内存排序的限制

为什么等值查询比多值查询更高效呢？

虽然都是“先搜后排”，但等值查询支持保持排序的情况下搜索，而后者不支持，需要额外内存排序。

(4) 哈希索引

哈希索引就是将字段的值进行哈希计算后作为索引，其强大之处在于实现时间复杂度为 $O(1)$ 查找，但对于经常需要排序或范围查询的集合不建议使用哈希索引。

3.5 聚合查询

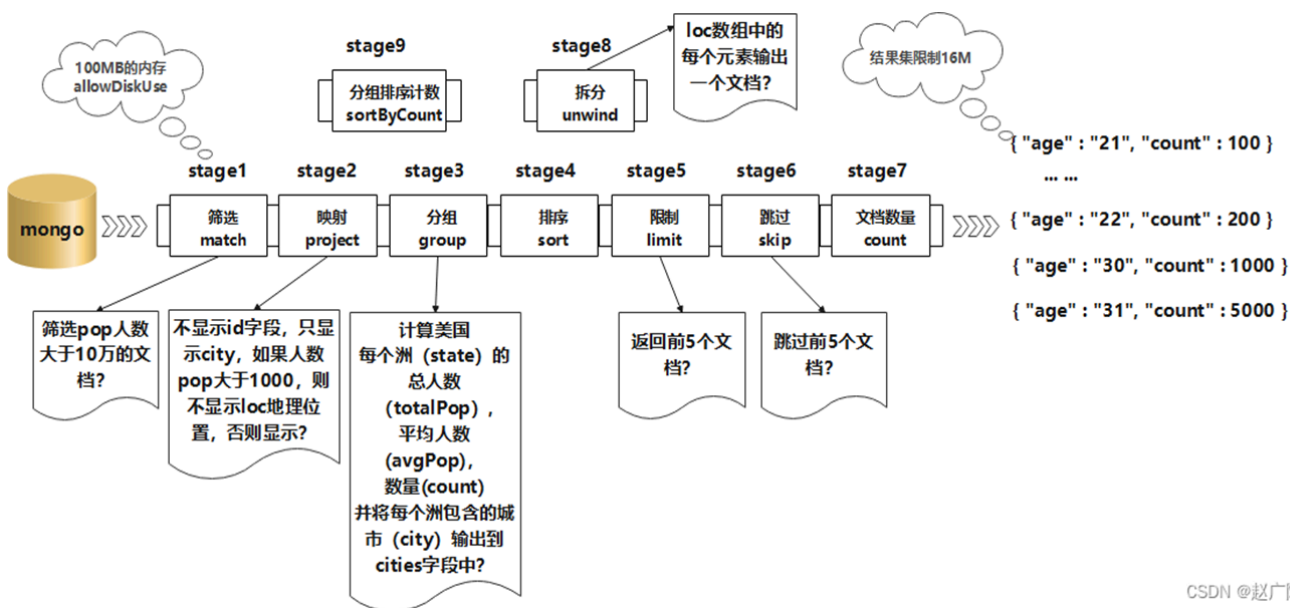
3.5.1 定义

转化合并由多个文档的数据来生成新的，主要用于**处理数据**（例如分组统计平均值、求和、最大值等），并返回计算后的数据结果。

方式：有两种方式计算聚合：**聚合管道**（Pipeline）和 **MapReduce**。前者快且简单，后者复杂。

3.5.2 流程

将文档输入处理管道，在管道内完成对文档的操作，最终将文档转换为聚合结果



CSDN @赵广陆

3.5.3 聚合使用

MongoDB中聚合的方法使用aggregate(), 基本语法格式：`db.collection.aggregate(pipeline, options)`

Pipeline：**数组**，一系列数据聚合操作或阶段。

Options：**文档**，可选，aggregate()传递给聚合命令的其他选项。2.6版中的新增功能：仅当将管道指定为数组时才可用。

\$match 匹配

\$group 指定分组格式

上面直接指定了几个操作，从题目中的实际应用来学习掌握，而不是死扣定义。

但其中group的使用语法要提前给出：

📢 Important

{ \$group: { _id: , : { : }, ... } }

参数说明：

id：必填，可以指定id值为null来为整个输入文档计算累计值。

field：剩余的计算字段是可选的，并使用运算符进行计算。

*_id*和表达式可以接受任何有效的表达式。

运算符有：

名称	描述
\$avg	计算均值
\$first	返回每组第一个文档，如果有排序，按照排序，如果没有按照默认的存储的顺序返回第一个文档。
\$last	返回每组最后一个文档，如果有排序，按照排序，如果没有按照默认的存储的顺序返回最后一个文档。
\$max	根据分组，获取集合中所有文档对应值的最大值。
\$min	根据分组，获取集合中所有文档对应值的最小值。
\$push	将指定的表达式的值添加到一个数组中。
\$addToSet	将表达式的值添加到一个集合中（无重复值，无序）。
\$sum	计算总和
\$stdDevPop	返回输入值的总体标准偏差（population standard deviation）
\$stdDevSamp	返回输入值的样本标准偏差（the sample standard deviation）

3.5.4 Mapreduce使用

map分解reduce合并

3.6 数据的导入与导出

数据导入导出的格式有CSV和JSON两种。

💡 Tip

导出：mongoexport -d dbname -c collectionname -o file --type json/csv -f field

•参数说明：

•-d：数据库名

•-c：collection名

•-o：输出的文件名

•--type：输出的格式，默认为json

•-f：输出的字段，如果-type为csv，则需要加上-f " 字段名 "

例如：mongoexport -d test -c users -o /mongoDB/users.json --type json -f " _id,name,age,status "

💡 Tip

导入：mongoimport -d dbname -c collectionname --file filename --headerline --type json/csv -f field

•参数说明：

•-d：数据库名

•-c：collection名

•--file：要导入的文件

•--type：导入的格式默认json

•-f：导入的字段名

•--headerline：如果导入的格式是csv，则可以使用第一行的标题作为导入的字段

例如：mongoimport -d test -c users --file /mongodb/articles.json --type json

3.7 MongoDB副本集

副本集（Replica Set）是一个有自动故障恢复功能的**主从集群**，MongoDB通过这种架构来实现数据的高可用性和容错性。每个副本集中只有一个主节点（读写），用于处理写操作，其他节点作为从节点（只读），负责同步主节点的数据。

当主节点未与集合中的其他成员通信超过配置的**选举超时时间（默认为10秒）**时，选出新的主节点上位。

📌 Note

rs.initiate(config);初始化配置：作用是选举主节点。**用于启动副本集**，并根据配置选举主节点

rs.status() **查看副本集状态**，确认主从节点配置是否成功

题目讲解

总的来讲，MongoDB部分就是记忆，并不难记，本文也给出了许多关键性总结技巧，同学们或可以练习头歌时对命令多加留意。

1.【单选题】如果在多个属性上创建一个复合索引，例如db.myColl.createIndex({ score: 1, price: 1, category: 1 }), 那么以下哪个查询无法从索引获益?

- A、db.myColl.find({ category:"apple", price:20, score:5 })
- B、db.myColl.find({ category:"apple", price:{\$gte:20, \$lte:30} })
- C、db.myColl.find({ category:"apple", score:{\$gte:4} })
- D、db.myColl.find({ score:{\$gte:4} })

我的答案: B

💡 Tip

释: 复合索引的查询优化是根据“索引键的顺序”和“前缀匹配原则”进行的。人话说就是，每个查询需要从索引定义的最左前缀开始利用。

所以，谁是最左前缀? score。谁没有score? B 结束

2.【单选题】在MongoDB中，创建一个数据库的命令是()

- A、alterB、createC、useD、new

我的答案: B

💡 Important

简要凝练一下，由大到小: use、drop数据库;

级别	操作
数据库	use (创建/切换数据库)、drop (删除数据库)
集合	create (创建集合)、drop (删除集合)
文档	insert (插入文档)、remove (删除文档)、update (更新文档)、find (查询文档)
索引	createIndex (创建索引)、dropIndex (删除索引)、ensureIndex (确保索引，后台【background】创建)

3.【单选题】()指定一个函数，在结果返回给客户端之前，每个分组都会执行一次这个函数。

- A、finalizeB、aggregateC、groupD、reduce

我的答案: A

4.【单选题】使用MongoDB的aggregate方法时，使用()参数在结果文档中插入值到一个数组中，但不创建副本。

- A、pushB、addToSetC、maxD、pull

我的答案: B

5.【单选题】在MongoDB客户端使用命令()来启动一个新的副本集。

- A、rs.initiate()B、rs.add()C、rs.start()D、rs.on()

我的答案: A

6.【单选题】在MongoDB中，当一个key有多个value时，需要使用数组，数组使用()括起来。

A、中括号B、双引号C、大括号D、小括号

我的答案: A

7.【单选题】()是在建立唯一索引时指定是否删除重复记录。

A、dropDupsB、nameC、backgroundD、unique

我的答案: A

8.【单选题】在MapReduce函数中,()参数表示统计函数

A、totalB、outC、queryD、reduce

我的答案: D

9.【单选题】在MongoDB中插入的文档一般为BSON的数据结构, BSON中, 值为**字符串**的需要使用()引起来

A、中括号B、大括号C、小括号D、双引号

我的答案: B

10.【单选题】在MongoDB中, 删除文档的命令是()

A、db.collectionName.drop()B、db.collectionName.delete()C、db.collectionName.remove()D、delete from

我的答案: C

11.【单选题】在聚合框架中,()操作符用于修改输入文档的结构。

A、\$limitB、\$skipC、\$matchD、\$project

我的答案: D

12.【单选题】MongoDB是由()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。

A、CB、JavaC、C++D、Python

我的答案: C

13.【单选题】MongoDB查询时, 使用()只查询符合条件的一个文档

A、select(1)B、findOneC、selectOneD、find(1)

我的答案: B

14.【单选题】在聚合框架中,()操作符用于过滤数据, 只输出符合条件的文档。

A、\$limitB、\$matchC、\$skipD、\$project

我的答案: B

15.【单选题】指定MongoDB**日志**文件的启动参数是()

A、logpathB、portC、bind_ipD、dbpath

我的答案: A

16.【单选题】在MongoDB中, 删除数据库的命令是()

A、db.removeDatabase()B、drop databaseC、db.dropDatabase()D、delete database

我的答案: C

17.【单选题】在MapReduce函数中, ()参数表示统计**结果**存放集合

A、total B、last C、out D、query

我的答案: C

18.【单选题】在MongoDB客户端查看副本集状态使用()命令

A、rs.initiate() B、rs.conf() C、rs.status() D、rs.add()

我的答案: C

19.【单选题】查询people集合中, 不同的年龄值的语句为()

A、db.runCommand({ " distinct " : " people " , " count " : " age " })

B、db.runCommand({ " count " : " people " , " key " : " age " })

C、db.runCommand({ " collection " : " people " , " key " : " age " })

D、db.runCommand({ " distinct " : " people " , " key " : " age " })

我的答案: D

20.【单选题】使用MongoDB的aggregate方法时, 使用()参数在结果文档中插入值到一个数组中, 但不创建副本。

A、push B、addToSet C、max D、pull

我的答案: B

21.【单选题】MongoDB中精确删除索引的基本语法格式为()

A、db.runCommand({dropIndexes:"collectionName",index:"indexName"})

B、db.runCommand({getIndexes:"collectionName",index:"indexName"})、

C、db.runCommand({deleteIndexes:"collectionName",index:"indexName"})

D、db.runCommand({removeIndexes:"collectionName",index:"indexName"})

我的答案: A

22.【单选题】指定服务**端口号**的启动参数是()

A、port B、bind_ip C、dbpath D、logpath

我的答案: A

23.【单选题】使用explain查询分析的结果集中, ()表示当前查询所需时间。

A、indexOnly B、indexBounds C、millis D、cursor

我的答案: C

24.【单选题】()参数的意思是抛出异常的级别

A、writeConcern B、multi C、upsert D、update

我的答案: A

25.【单选题】建索引过程会阻塞其它数据库操作, ()可指定以后台方式创建索引

A、background B、name C、unique D、dropDups

我的答案: A

26.【单选题】指定数据库**路径**的启动参数是()

A、logpath B、dbpath C、bind_ip D、port

我的答案: B

27.【单选题】**绑定服务IP**的启动参数是()

A、bind_ip B、dbpath C、logpath D、port

我的答案: A

28.【单选题】使用的比较运算符进行MongoDB查询时, 小于号表示为()符号。

A、\$gte B、\$gt C、\$lte D、\$lt

我的答案: D

Important

```
$lt    <    (less than )  
  
$lte   <=   (less than or equal to )  
  
$gt    >    (greater than )  
  
$gte   >=   (greater than or equal to)
```

29.【单选题】MongoDB创建索引时, ()为指定按升序创建索引。

A、true B、-1 C、1 D、false

我的答案: C

30.【单选题】MongoDB使用()方法来创建索引。

A、ensureIndex B、create Index C、newIndex D、addIndex

我的答案: A

31.【单选题】在MongoDB中删除文档时, ()参数如果设为 true 或 1, 则只删除一个文档。

A、removeOne B、justOne C、query D、One

我的答案: B

32.【单选题】在MongoDB中, 插入文档的命令是()

A、db.collectionName.insert() B、insert into C、db.collectionName.add() D、add into

我的答案: A

33.【单选题】在MongoDB分片中, ()用于存储实际的数据块。

A、Shard B、Config Server C、Query Routers D、replSet

我的答案: A

34.【单选题】MongoDB服务的默认端口号是() (单选题)

A、8088B、8080C、27017D、27010

我的答案: C

35.【单选题】在MongoDB中, 删除库中的集合的命令是()

A、db.colletionName.remove()B、db.collectionName.drop()C、db.colletionName.delete()D、delete from

我的答案: B

36.【单选题】()操作提供了查询信息, 使用索引及查询统计等。

A、skipB、hintC、explainD、sort

我的答案: C

37.【单选题】在MongoDB中, 更新文档的命令是()

A、db.collectionName.modify()B、db.collectionName.find()C、db.collectionName.update()D、db.collectionName.save()

我的答案: C

38.【单选题】在MapReduce函数中, ()参数表示一个筛选条件

A、onB、queryC、whereD、map

我的答案: B

39.【单选题】在MongoDB中, 查询文档的命令是()

A、db.collectionName.insert()B、db.collectionName.update()C、select fromD、db.collectionName.find()

我的答案: D

40.【单选题】进行批量修改时, 一定不能使用强硬更新操作, 需要加入()操作符。

A、\$upsertB、\$unsetC、\$incD、\$set

我的答案: D

41.【单选题】在聚合框架中, ()操作符用来限制MongoDB聚合管道返回的文档数。

A、\$projectB、\$limitC、\$skipD、\$match

我的答案: B

42.【多选题】mongodb复制中, 各个节点常见的搭配方式有()

A、两主一从B、一主一从C、一主多从D、两主多从

我的答案: BC

“学友”工作室出品

更多学科复习资料、竞赛学习等内容

欢迎关注华理“以学会友ing”公众号