

MongoDB Quickref Sheet

Admin Commands

```
help
db.help()

show dbs
show databases
use <db>
show collections
```

aggregate \$ keywords ^{1) 2)}

```
$group
$avg
$min
$max
$sum
$first
$last
$lookup
$project
```

DB Commands

```
db.collection.save()
db.collection.find()
db.colle...find().pretty()
db.collection.findOne()

db.collection.insert()
db.collection.update()
```

Sorting ³⁾

```
cursor.sort()
```

find/update \$ Keywords

```
$and: [{},{}]
$or: [{},{}]
$in: [1,2,3,7]
$set: [{},{}]
$unset: {field:1}
$rename: {"old": "new"}

$exists: true/false
$gt: 1
$gte: 1
$lt: 1
$lte: 1
```

Import/Export data

```
// import csv into empty collection
mongoimport --db userdb --collection user --type csv --headerline --file C:\Use...\users.csv
```

```
// add csv to existing collection
mongoimport --db userdb --collection users --type csv --headerline --mode merge --file
C:\Use...\sex.csv
```

```
// import json into empty collection
mongoimport --db proj --collection docs --type json --file
C:\Users\121988\Documents\52553\mongo.json
```

```
//export an existng collection
mongoexport.exe /db:userdb /collection:users /jsonArray /pretty /out:users.json
```

Find Examples

find(query)

```
db.user.find({age:{$gt:19}})
```

```
db.user.find({$and: [{age:{$gt:19}}, {carReg: {$exists:true}}]})
```

find(query, projection)

```
db.User.find({age: {$gt: 20}}, {email:1})
```

```
//Return only the first_name and surname attributes of all documents
```

```
db.User.find({}, {_id:false, first_name:1, surname:1})
```

```
db.student.find({address:{$exists: true}}, {_id:0, "address.county":1})
```

Update Examples

Note the use of \$set and \$unset

```
db.users.update({_id:105},{$set:{"carReg":"161-M0-4"}})
```

```
db.users.update({$and: [{Sex:"M"},{age:{$gt:20}}]},{$set:{"title":"Mr."},{multi:true})
```

```
db.users.update({_id:{$in:[101,103,107]}},{$set:{carReg:""}},{multi:true})
```

```
db.users.update({_id:{$in:[101,103,107]}},{$unset:{carReg:1}},{multi:true})
```

```
db.users.updateMany({},{$rename:{"fname":"Name"}})
```

Aggregate Examples

db.collection.aggregate ⁴⁾ ([\$match, \$group, \$sort])

```
db.collection.aggregate([{$match:{status:"A"}},{$group: {_id:"$cust_id",total:{$sum:"$amount"}}},{$sort:{total:-1}}])
```

\$group

\$group same as Group by in **MySQL**

```
//Get the average gpa for all students
db.users.aggregate([{$group: {_id:null, Average:{$avg:"$gpa"}}}])
```

```
//Get the Maximum GPA per age group
db.march8.aggregate([{$group: {_id:"$age", "Max GPA per Age":{$max:"$gpa"}}}])
```

```
//To sort: $sort
db.march8.aggregate([{$group: {_id:"$age", "Max GPA per Age":{$max:"$gpa"}}}, {$sort:{_id:1}}])
```

\$lookup

\$lookup: {from, localField, foreignField, as}

from:	<collection to join>
localField:	<field from the input documents>
foreignField:	<field from the documents of the "from" collection>
as:	<output array field>

```
// .aggregate([{$lookup}])
db.docs.aggregate([{$lookup:{from:"docs", localField:"modules", foreignField: "_id",
```

```
as: "Details" } } ] )
```

```
// .aggregate([{$match},{ $lookup}])
db.cityinfo.aggregate([{$match:{name:"Tom"}},{ $lookup:{from:"cityinfo", localField:"addresses",
foreignField: "_id", as:"Details"}}])
```

```
// .aggregate([{$match},{ $lookup},{ $project}])
db.cityinfo.aggregate([{$match:{city:"CHESTERFIELD"}},{ $lookup:{from:"cityinfo",
localField:"state", foreignField:"_id", as:"StateInfo"}},{ $project:{capital:0}}])
```

Query, Update, Options

db.collection.update⁵⁾(query, update, options)

```
db.collection.update(
  <query>,
  <update>,
  {
    upsert: <boolean>,
    multi: <boolean>,
    writeConcern: <document>,
    collation: <document>,
    arrayFilters: [ <filterdocument1>, ... ]
  }
)
```

Indexing

```
db.collection.getIndexes()
```

```
db.user.createIndex({age:1})
```

```
db.collection.dropIndex({age:1})
```

Note: The index on **_id** cannot be dropped

1)
https://docs.mongodb.com/manual/reference/operator/aggregation/group/#pipe._S_group

2)
<https://docs.mongodb.com/manual/reference/operator/aggregation/lookup/>

3)
<https://docs.mongodb.com/manual/reference/method/cursor.sort/>

4)
<https://docs.mongodb.com/manual/reference/method/db.collection.aggregate/>

5)
<https://docs.mongodb.com/manual/reference/method/db.collection.update/#db.collection.update>

From:

<http://www.hdip-data-analytics.com/> - HDip Data Analytics

Permanent link:

<http://www.hdip-data-analytics.com/help/mongodb/refsheet>

Last update: 2020/06/20 14:39

