MongoDB Cheat Sheet

Ondrej Sika <ondrej@ondrejsika.com>

Database

Select Database

```
use DATABASE_NAME
```

Example

```
> use mydb
switched to db mydb
```

Check selected database

```
> show dbs
admin  0.000GB
local  0.000GB
test   0.000GB
```

Create Database

Create database by insert of first document.

```
switched to db mydb
> db.pets.insert({name: 'Pista', kind: 'dog'})
WriteResult({ "nInserted" : 1 })
> show dbs
admin  0.000GB
local  0.000GB
mydb   0.000GB
test   0.000GB
```

Drop Database

```
> db.dropDatabase()
{ "dropped" : "mydb", "ok" : 1 }
> show dbs
admin  0.000GB
local  0.000GB
test   0.000GB
```

Collection

Create Collection

```
> use test
switched to db test
> db.createCollection("mycol")
{ "ok" : 1 }
> db.createCollection("crappedcol", {
    capped: true,
    autoIndexID: true,
    size: 1024,
    max: 1
})
{ "ok" : 1 }
```

List collections

```
> show collections
mycol
crappedcol
```

Drop Collection

```
> db.mycol.drop()
true
```

Insert Document

```
> db.COLLECTION_NAME.insert(DOCUMENT)
```

Example

```
db.pets.insert({
    name: 'Fista',
    kind: 'cat',
    age: 2,
    colors: ['black', 'white'],
})
db.pets.insert({
    name: 'Pista',
    kind: 'dog',
    age: 4,
    colors: ['brown'],
})
db.pets.insert({
    name: 'Ben',
    kind: 'dog',
    age: 12,
    colors: ['white'],
})
db.pets.insert({
    name: 'Mista',
    kind: 'rat',
    age: 1,
    colors: ['black', 'brown'],
})
```

Query Document

Find

Select all documents

```
> db.COLLECTION_NAME.find()
```

or

```
> db.COLLECTION_NAME.find({})
```

Example

```
> db.pets.find()
{ "_id" : ObjectId("58dc747ad3fbf12faaaa1706"), "name" : "Fista", "kind" : "cat", "age" : 2, "colors" : [ "black", "white" ] }
{ "_id" : ObjectId("58dc74e4d3fbf12faaaa1707"), "name" : "Pista", "kind" : "dog", "age" : 4, "colors" : [ "brown" ] }
{ "_id" : ObjectId("58dc74e4d3fbf12faaaa1708"), "name" : "Mista", "kind" : "rat", "age" : 1, "colors" : [ "black", "brown" ] }
```

For pretty formating, use `.pretty()`

```
> db.pets.find().pretty()
{
    "_id" : ObjectId("58dc747ad3fbf12faaaa1706"),
    "name" : "Fista",
    "kind" : "cat",
    "age" : 2,
    "colors" : [
        "black",
        "white"
    ]
}
...
```

Operators

Operation	Syntax	Example	SQL
Equality	`{<key>:<value>}`	`{kind: 'rat'}`	`where kind = 'rat'`
Less Than	`{<key>:{$lt:<value>}}`	`{age: {$lt: 2}}`	`where age < 2`
Less Than Equals	`{<key>:{$lte:<value>}}`	`{age: {$lte: 2}}`	`where age <= 2`
Greater Than	`{<key>:{$gt:<value>}}`	`{age: {$gt: 2}}`	`where age > 2`
Greater Than Equals	`{<key>:{$gte:<value>}}`	`{age: {$gte: 2}}`	`where age >= 2`
Not Equals	`{<key>:{$ne:<value>}}`	`{age: {$ne: 2}}`	`where age != 2`
In	`{<key>:{$in:[<value1>, <value2>, ...]}}`	`{age: {$in: [1, 2, 3]}}`	`where age in (1, 2, 3)`

Examples

```
> db.pets.find({name: 'Pista'})
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170a"), "name" : "Pista", "kind" : "dog", "age" : 4, "colors" : [ "brown" ] }
> db.pets.find({kind: 'dog'})
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170a"), "name" : "Pista", "kind" : "dog", "age" : 4, "colors" : [ "brown" ] }
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170b"), "name" : "Ben", "kind" : "dog", "age" : 12, "colors" : [ "white" ] }
> db.pets.find({colors: 'black'})
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170c"), "name" : "Mista", "kind" : "rat", "age" : 1, "colors" : [ "black", "brown" ] }
{ "_id" : ObjectId("58dc7f6cd3fbf12faaaa170d"), "name" : "Fista", "kind" : "cat", "age" : 2, "colors" : [ "black", "white" ] }
> db.pets.find({colors: {$ne: 'black'}})
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170a"), "name" : "Pista", "kind" : "dog", "age" : 4, "colors" : [ "brown" ] }
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170b"), "name" : "Ben", "kind" : "dog", "age" : 12, "colors" : [ "white" ] }
```

Columns

```
db.COLLECTION_NAME.find(QUERY, COLLUMNS)
```

Example

```
> db.pets.find({}, {name: 1})
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170a"), "name" : "Pista" }
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170b"), "name" : "Ben" }
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170c"), "name" : "Mista" }
{ "_id" : ObjectId("58dc7f6cd3fbf12faaaa170d"), "name" : "Fista" }
> db.pets.find({}, {name: 1, _id: 0})
{ "name" : "Pista" }
{ "name" : "Ben" }
{ "name" : "Mista" }
{ "name" : "Fista" }
> db.pets.find({}, {_id: 0})
{ "name" : "Pista", "kind" : "dog", "age" : 4, "colors" : [ "brown" ] }
{ "name" : "Ben", "kind" : "dog", "age" : 12, "colors" : [ "white" ] }
{ "name" : "Mista", "kind" : "rat", "age" : 1, "colors" : [ "black", "brown" ] }
{ "name" : "Fista", "kind" : "cat", "age" : 2, "colors" : [ "black", "white" ] }
```

AND

```
db.COLLECTION_NAME.find({key1:value1, key2:value2})
```

Example

```
> db.pets.find({age: {$gte: 2}, kind: 'dog'})
{ "_id" : ObjectId("58dc74e4d3fbf12faaaa1707"), "name" : "Pista", "kind" : "dog", "age" : 4, "colors" : [ "brown" ] }
```

OR

```
db.COLLECTION_NAME.find({$or: [{key1: value1}, {key2:value2}]})
```

Example

```
> db.pets.find({$or: [{kind: 'rat'}, {kind: 'cat'}]})
{ "_id" : ObjectId("58dc747ad3fbf12faaaa1706"), "name" : "Fista", "kind" : "cat", "age" : 2, "colors" : [ "black", "white" ] }
{ "_id" : ObjectId("58dc74e4d3fbf12faaaa1708"), "name" : "Mista", "kind" : "rat", "age" : 1, "colors" : [ "black", "brown" ] }
```

Limit & Offset

```
> db.pets.find().limit(1)
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170a"), "name" : "Pista", "kind" : "dog", "age" : 4, "colors" : [ "brown" ] }
> db.pets.find().skip(1)
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170b"), "name" : "Ben", "kind" : "dog", "age" : 12, "colors" : [ "white" ] }
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170c"), "name" : "Mista", "kind" : "rat", "age" : 1, "colors" : [ "black", "brown" ] }
{ "_id" : ObjectId("58dc7f6cd3fbf12faaaa170d"), "name" : "Fista", "kind" : "cat", "age" : 2, "colors" : [ "black", "white" ] }
> db.pets.find().skip(1).limit(1)
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170b"), "name" : "Ben", "kind" : "dog", "age" : 12, "colors" : [ "white" ] }
```

Sort

```
> db.pets.find().sort({name: 1})
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170b"), "name" : "Ben", "kind" : "dog", "age" : 12, "colors" : [ "white" ] }
{ "_id" : ObjectId("58dc7f6cd3fbf12faaaa170d"), "name" : "Fista", "kind" : "cat", "age" : 2, "colors" : [ "black", "white" ] }
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170c"), "name" : "Mista", "kind" : "rat", "age" : 1, "colors" : [ "black", "brown" ] }
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170a"), "name" : "Pista", "kind" : "dog", "age" : 4, "colors" : [ "brown" ] }
> db.pets.find().sort({name: -1} )
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170a"), "name" : "Pista", "kind" : "dog", "age" : 4, "colors" : [ "brown" ] }
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170c"), "name" : "Mista", "kind" : "rat", "age" : 1, "colors" : [ "black", "brown" ] }
{ "_id" : ObjectId("58dc7f6cd3fbf12faaaa170d"), "name" : "Fista", "kind" : "cat", "age" : 2, "colors" : [ "black", "white" ] }
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170b"), "name" : "Ben", "kind" : "dog", "age" : 12, "colors" : [ "white" ] }
>
```

Update Documents

```
db.COLLECTION_NAME.update(SELECTIOIN_CRITERIA, UPDATED_DATA)
```

Example

```
> db.pets.update({name: 'Ben'}, {$set: {age: 13}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.pets.find({name: 'Ben'})
{ "_id" : ObjectId("58dc7af7d3fbf12faaaa1709"), "name" : "Ben", "kind" : "dog", "age" : 13, "colors" : [ "white" ] }
```

By default, MongoDB will update only a single document. To update multiple documents, you need to set a parameter 'multi' to true.

```
db.COLLECTION_NAME.update(SELECTIOIN_CRITERIA, UPDATED_DATA, {multi: true})
```

Example

```
> db.pets.update({name: 'Ben'}, {$set: {age: 13}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.pets.find({name: 'Ben'})
{ "_id" : ObjectId("58dc7af7d3fbf12faaaa1709"), "name" : "Ben", "kind" : "dog", "age" : 13, "colors" : [ "white" ] }
```

Save

Replace document by ID

```
db.COLLECTION_NAME.save({_id:ObjectId(),NEW_DATA})
```

Example

```
> db.pets.find({_id: ObjectId('58dc7af7d3fbf12faaaa1709')})
{ "_id" : ObjectId("58dc7af7d3fbf12faaaa1709"), "name" : "Ben", "kind" : "dog", "age" : 13, "colors" : [ "white" ] }
> db.pets.save({ "_id" : ObjectId("58dc7af7d3fbf12faaaa1709"), "name" :
> db.pets.save({ "_id" : ObjectId("58dc7af7d3fbf12faaaa1709"), "name" : "Benik", "kind" : "dog", "age" : 11 })
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.pets.find({_id: ObjectId('58dc7af7d3fbf12faaaa1709')})
{ "_id" : ObjectId("58dc7af7d3fbf12faaaa1709"), "name" : "Benik", "kind" : "dog", "age" : 11 }
```

Deleting Documents

```
db.COLLECTION_NAME.remove(DELLETION_CRITTERIA)
```

Remove only one

```
db.COLLECTION_NAME.remove(DELLETION_CRITTERIA, 1)
```

Example

```
> db.pets.find()
{ "_id" : ObjectId("58dc747ad3fbf12faaaa1706"), "name" : "Fista", "kind" : "cat", "age" : 2, "colors" : [ "black", "white" ] }
{ "_id" : ObjectId("58dc74e4d3fbf12faaaa1707"), "name" : "Pista", "kind" : "dog", "age" : 4, "colors" : [ "brown" ] }
{ "_id" : ObjectId("58dc74e4d3fbf12faaaa1708"), "name" : "Mista", "kind" : "rat", "age" : 1, "colors" : [ "black", "brown" ] }
{ "_id" : ObjectId("58dc7af7d3fbf12faaaa1709"), "name" : "Benik", "kind" : "dog", "age" : 11 }
> db.pets.remove({age: {$gt: 1}}, 1)
WriteResult({ "nRemoved" : 1 })
> db.pets.remove({age: {$gt: 1}})
WriteResult({ "nRemoved" : 2 })
> db.pets.find()
{ "_id" : ObjectId("58dc74e4d3fbf12faaaa1708"), "name" : "Mista", "kind" : "rat", "age" : 1, "colors" : [ "black", "brown" ] }
```

Delete all documents

```
db.COLLECTION_NAME.remove({})
```

Example

```
> db.pets.remove({})
WriteResult({ "nRemoved" : 1 })
```

Indexing

Get indexes

```
db.COLLECTION.getIndexes()
```

Example

```
> db.pets.getIndexes()
[
    {
        "v" : 1,
        "key" : {
            "_id" : 1
        },
        "name" : "_id_",
        "ns" : "test.pets"
    }
]
```

Create index

```
db.COLLECTION_NAME.ensureIndex({KEY:1})
```

or multi key index

```
db.COLLECTION_NAME.ensureIndex({KEY1:1, KEY2:1})
```

Examples:

```
> db.pets.ensureIndex({age:1})
{
    "createdCollectionAutomatically" : false,
    "numIndexesBefore" : 1,
    "numIndexesAfter" : 2,
    "ok" : 1
}
```

Parameters:

```
db.COLLECTION_NAME.ensureIndex(INDEX, PARAMS)
```

Eg.:

```
db.COLLECTION_NAME.ensureIndex(INDEX, {
    unique: 1,
    backgroud: 1,
})
```

Background means build index on background, don't block the DB.

Example:

```
> db.pets.ensureIndex({name: 1}, {
    unique: 1,
    backgroud: 1,
})
> db.pets.find()
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170a"), "name" : "Pista", "kind" : "dog", "age" : 4, "colors" : [ "brown" ] }
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170b"), "name" : "Ben", "kind" : "dog", "age" : 12, "colors" : [ "white" ] }
{ "_id" : ObjectId("58dc7f1fd3fbf12faaaa170c"), "name" : "Mista", "kind" : "rat", "age" : 1, "colors" : [ "black", "brown" ] }
{ "_id" : ObjectId("58dc7f6cd3fbf12faaaa170d"), "name" : "Fista", "kind" : "cat", "age" : 2, "colors" : [ "black", "white" ] }
> db.pets.insert({name: 'Pista', kind: 'bat'})
WriteResult({
    "nInserted" : 0,
    "writeError" : {
        "code" : 11000,
        "errmsg" : "E11000 duplicate key error collection: test.pets index: name_1 dup key: { : \"Pista\" }"
    }
})
```

Drop index

```
db.COLLECTION_NAME.dropIndex({KEY:1})
```

Example

```
> db.pets.dropIndex({age:1})
{ "nIndexesWas" : 3, "ok" : 1 }
```

Count

```
db.COLLECTION_NAME.count()
```

or

```
db.COLLECTION_NAME.find(QUERY).count()
```

Example

```
> db.pets.count()
4
> db.pets.find({kind: 'dog'})
2
```

Aggregation

aggregation operators

- `$sum`
- `$avg`
- `$min`
- `$max`
- `$push`
- `$addToSet`
- `$first`
- `$last`

Example

```
> db.pets.aggregate([{$group : {_id : "$kind", count: {$sum : 1}}}])
{ "_id" : "cat", "count" : 1 }
{ "_id" : "rat", "count" : 1 }
{ "_id" : "dog", "count" : 2 }
> db.pets.aggregate([{$group : {_id : "$kind", count: {$sum : 1}, age: {$avg: '$age'}}}])
{ "_id" : "cat", "count" : 1, "age" : 2 }
{ "_id" : "rat", "count" : 1, "age" : 1 }
{ "_id" : "dog", "count" : 2, "age" : 8 }
```

By multiple keys

```
> db.pets.aggregate([{$group : {_id : {kind: '$kind', age: '$age'}, count: {$sum : 1}}}])
{ "_id" : { "kind" : "cat", "age" : 2 }, "count" : 1 }
{ "_id" : { "kind" : "rat", "age" : 1 }, "count" : 1 }
{ "_id" : { "kind" : "dog", "age" : 12 }, "count" : 1 }
{ "_id" : { "kind" : "dog", "age" : 4 }, "count" : 1 }
```