

Database Administration

Command line tools

mongo | Start the shell from the command line
 command line options:
 -host hostname | hostname to connect to
 -port 27017 | connect to port 27017 (default)
 -u foo | username foo
 -p bar | password bar
 --authenticationDatabase arg | database to authenticate to
 mongoimport | import data from a file to mongod
 mongodump | dump contents of a database to files
 mongorestore | restore contents of a dump to a database
 mongotop | profile the resources mongo is consuming
 mongostat | profile the amount of time spent in collections

Basic shell commands

help | get help for the context you're in
 quit() | exit the shell
 db | show the selected database
 use foo | select and use the database foo
 show dbs | show databases on server
 show collections | show collections in the current db
 show users | show the users in the current db
 show roles | list all built in and user defined roles in the current db

Collection commands (db.collection.)

help() | show a list of help commands for a collection
 copyTo() | copy a collection to a new collection name
 count() | get number of documents in the collection
 drop() | remove the collection from the database
 mapReduce() | performs map-reduce data aggregate
 renameCollection() | rename a collection
 stats() | get stats about the collection

Indexing

dropIndex() | removes a specified index on a collection
 createIndex() | creates an index if it does not currently exist
 getIndexes() | gets details on the indexes on a collection
 reIndex() | rebuilds all existing indexes on a collection
 compact | defragments a collection and rebuilds the indexes

Index options

Index Properties
 expireAfterSeconds | delete documents after set time
 unique | create indexes that allow only unique data
 sparse | index only documents having the index field
 partial | index only documents meeting a filter criteria
Index Creation Options
 background | create index in the background

Cursors (db.collection.find.)

iterate() | iterate on a cursor using its variable reference
 count() | return a count of the documents in a cursor
 explain() | get the query execution plan for a cursor
 hasNext() | true if cursor has documents and can be iterated
 hint() | force db to use a specific index for a query
 limit() | constrain the size of a cursor's result set
 next() | return the next document in a cursor
 skip() | skip through some documents and then return results
 sort() | return results ordered according to a sort specification
 toArray() | return an array of all documents for the cursor
 pretty() | pretty print the documents returned

System Administration

DB commands (db.)

help() | show a list of help commands for a db
 copyDatabase() | copies a db to another db
 dropDatabase() | remove the current db
 getLastError() | get status of last error
 hostInfo() | get info about the host system
 serverStatus() | get an overview of server status
 shutdownServer() | shutdown current server
 stats() | get stats on the current db selected
 version() | get the current version of the server

Authentication

db.createUser() | create a user in the system.users collection
 db.changeUserPassword() | change an existing users password
 db.dropUser() | remove a user from a database
 db.auth() | authenticates a user to a database
 db.logout() | logout from a database

CRUD Queries

Query commands (db.collection.)

find() | perform a query and return a results cursor
 findAndModify() | atomically find and update
 findOne() | perform a query and return a document
 distinct() | query for distinct documents on a field
 insert() | insert a new document into the collection
 remove() | remove a document from a collection
 save() | wrapper around insert and update to save
 update() | update one or more documents

Query Operators (\$)

Comparison
 \$gt | matches values greater than the value
 \$gte | matches values greater than or equal the value
 \$in | matches values supplied in an array
 \$lt | matches values less than the value
 \$lte | matches values less than or equal the value
 \$ne | matches all values that are not equal to given
 \$nin | matches values that do not exist in an array
Logical
 \$or | joins query clauses with a logical OR
 \$and | joins query clauses with a logical AND
 \$not | returns documents that do not match
 \$nor | joins query clauses with a logical NOR
Element
 \$exists | matches documents that have a field
 \$type | matches a field if it is of a given type
Evaluation
 \$mod | perform a modulo on a field and select if 0
 \$regex | matches a regex expression on a field
 \$where | matches against a JavaScript expression
Geospatial
 \$geoWithin | matches within a bounding geometry
 \$geoIntersects | matches intersection in a geometry
 \$near | matches near a geospatial point
 \$nearSphere | matches near a point on a sphere
Array
 \$all | matches arrays that contain all elements given
 \$elemMatch | matches multiple conditions in array
 \$size | matches if the array is of specified size

Update Operators (\$)

Fields
 \$inc | increment a value by a specified amount
 \$rename | rename a field
 \$setOnInsert | set a value only if inserting
 \$set | set the value of a field on an existing document
 \$unset | remove the field from an existing document
Array Operators
 \$ | update the first element in an array that matches
 \$addToSet | add element to array if it doesn't exist
 \$pop | remove the first or last item of an array
 \$pullAll | remove multiple values from an array
 \$pull | remove items which match a query statement
 \$push | adds an item to an array
Array Modifiers
 \$each | modify \$push and \$addToSet to add many
 \$slice | modify \$push to limit size of updated array
 \$sort | modify \$push to reorder documents in array
 \$position | modify \$push to specify the location to push at in array.
Bitwise
 \$bit | performs bitwise AND and OR updates
Isolation
 \$isolated | improve isolation of the operation

Projection Operators (\$)

\$ | project the first element in an array that matches
 \$elemMatch | project only the first element match
 \$slice | limit number of elements projected from array

Analytic Queries

Aggregation Operators (\$)

Pipeline Operators
 \$project | reshape a document
 \$match | match documents against a query
 \$limit | restrict the number of documents returned
 \$skip | skip over some documents and return the rest
 \$unwind | open elements of an array into documents
 \$group | group on a field and aggregate values
 \$sort | sort on a specified field
 \$geoNear | get documents near a geospatial point

Aggregation Expression Operators (\$)

Accumulators (\$group and \$project stages)
 \$addToSet | return a unique array of values for group
 \$first | return the first value in a group
 \$last | return the last value in a group
 \$max | return the highest value in a group
 \$min | return the lowest value in a group
 \$avg | return an average of all values in a group
 \$push | return an array of values for a grouped field
 \$sum | return the sum of all values in a group
 \$stdDevPop | return the population standard deviation of the input
 \$stdDevSamp | return the sample standard deviation of the input
Boolean Operators
 \$and | returns true when all values in array are true
 \$or | returns true when any value in its array are true
 \$not | returns boolean value that is opposite of input
Comparison Operators
 \$cmp | return the result of a compare as an integer
 \$eq | return true if two values are equal
 \$gt | return true if first value greater than 2nd
 \$gte | return true if first value greater or equal to 2nd
 \$lt | return true if first value less than 2nd
 \$lte | return true if first value less than or equal 2nd
 \$ne | return true if two values are not equal
Arithmetic Operators
 \$add | return the sum of an array of numbers
 \$divide | return the result of dividing two numbers
 \$mod | return the modulo of dividing two numbers
 \$multiply | return the product of an array of numbers
 \$subtract | return the result of subtracting 2 numbers
String Operators
 \$concat | concatenate two strings
 \$strcasecmp | return an int reflecting a comparison
 \$substr | return a portion of a string
 \$toLower | convert a string to lowercase
 \$toUpper | convert a string to uppercase
Date Operators
 \$dayOfYear | return an int between 1 and 366
 \$dayOfMonth | return an int between 1 and 31
 \$dayOfWeek | return an int between 1 and 7
 \$year | return the full year from a date
 \$month | return an int between 1 and 12
 \$week | return an int between 0 and 53
 \$hour | return an int between 0 and 23
 \$minute | return an int between 0 and 59
 \$second | return an int between 0 and 60
 \$millisecond | return millisecond portion of a date
 \$dateToString | return the date as a formatted string
Conditional Expressions
 \$cond | ternary style operator, takes 3 expressions
 \$ifNull | eval 1st expression, if null eval 2nd, return

Misc links

Release Notes
 Backup and Recovery
 Manage Journaling
 MongoDB Scripting
 Analyze Performance and Profiling
 Security
 Community

Replication (rs.)

add() | adds a member to a replica set
 addArb() | adds an arbiter to a replica set
 conf() | returns the replica set config document
 freeze() | prevents a member from becoming primary
 help() | get basic help for replica set functions
 initiate() | initializes a new replica set
 reconfig() | reconfigure a replica set with a new config
 remove() | remove a member from a replica set
 slaveOk() | allow reads to happen on a secondary
 status() | return a document with status of replica set
 stepDown() | force primary to step down
 syncFrom() | specify the member to sync from
Read Preferences
 primary | read only from the primary in a replica set
 primaryPreferred | prefer the primary but can read from secondary
 secondary | read only from a secondary in a replica set
 secondaryPreferred | prefer a secondary, read from primary last
 nearest | read from the nearest member in a replica set

Sharding (sh.)

addShard() | add a shard to the cluster
 addShardTag() | associate a shard with a tag
 addTagRange() | associate range of shard keys with tag
 disableBalancing() | disable balancing on a collection
 enableBalancing() | re-enable balancing on a collection
 enableSharding() | enables sharding on a database
 getBalancerHost() | get the mongos doing balancing
 getBalancerState() | true if the balancer is enabled
 help() | returns help text for the sh methods
 isBalancerRunning | true if the balancer is migrating
 moveChunk() | migrates a chunk in a sharded cluster
 removeShardTag() | disassociate a shard with a tag
 setBalancerState() | enable or disable the balancer
 shardCollection() | enables sharding for a collection
 splitAt() | divide a chunk in 2 based on shard key value
 splitFind() | divide a chunk in half based on a query
 startBalancer() | enable balancer and wait until started
 status() | reports on the status of a sharded cluster
 stopBalancer() | stop balancer and wait until stopped