# **Oblig 3 Databaser og nettverk 18.11.2021 kl. 12:00**

24.11.2021

1. **MongoDB**   
   I oppgave 1 trenger du data fra oppgave 1 i Oblig1.
   1. Opprett en MongoDB database **Oblig3** med en dokumentsamling (**Collection**) **Film**.

use oblig3;  
db.createCollection('film');

* 1. Lag **JSON**-objekter og sett inn data med følgende nøkler:  
     **FilmNr, Tittel, År, Land, Sjanger, Alder, Tid, Pris, Antall**Selve datene tar dere fra **Oblig1**, **oppgave 1**. Ingen **Pris** er uten verdi, **Antall** angir filmer på lager og kan godt ha verdien **0**.

db.film.remove({});

db.film.**insertMany**([  
{FilmNr: 1, Tittel: 'Casablanca', År: 1942, Land: 'USA', Sjanger: 'Drama',  
 Alder: 15, Tid: 102, Pris: 149.00, Antall: 3},  
{FilmNr: 2, Tittel: 'Fort Apache', År: 1948, Land: 'USA', Sjanger: 'Western',  
 Alder: 15, Tid: 127, Pris: 144.00, Antall: 0},  
{FilmNr: 3, Tittel: 'Apocalypse Now', År: 1979, Land: 'USA', Sjanger: 'Action',  
 Alder: 18, Tid: 155, Pris: 123.00, Antall: 2},  
{FilmNr: 4, Tittel: 'Streets of Fire', År: 1984, Land: 'USA', Sjanger: 'Action',  
 Alder: 15, Tid: 93, Pris: 124.00, Antall: 0},  
{FilmNr: 5, Tittel: 'High Noon', År: 1952, Land: 'USA', Sjanger: 'Western',  
 Alder: 15, Tid: 85, Pris: 123.00, Antall: 2},  
{FilmNr: 6, Tittel: 'Cinema Paradiso', År: 1988, Land: 'Italia', Sjanger: 'Komedie',  
 Alder: 11, Tid: 111, Pris: 123.00, Antall: 2},  
{FilmNr: 7, Tittel: 'Asterix hos britene', År: 1988, Land: 'Frankrike', Sjanger: 'Tegnefilm',  
 Alder: 7, Tid: 78, Pris: 149.00, Antall: 4},  
{FilmNr: 8, Tittel: "Veiviseren", År: 1987, Land: "Norge", Sjanger: "Action",  
 Alder: 15, Tid: 96, Pris: 87.00, Antall: 2},  
{FilmNr: 9, Tittel: 'Salmer fra kjøkkenet', År: 2001, Land: 'Norge', Sjanger: 'Komedie',  
 Alder: 7, Tid: 80, Pris: 149.00, Antall: 1},  
{FilmNr: 10, Tittel: 'Anastasia', År: 1997, Land: 'USA', Sjanger: 'Tegnefilm',  
 Alder: 7, Tid: 94, Pris: 123.00, Antall: 3},  
{FilmNr: 11, Tittel: 'La Grande bouffe', År: 1973, Land: 'Frankrike', Sjanger: 'Drama',  
 Alder: 15, Tid: 129, Pris: 87, Antall: 1},  
{FilmNr: 12, Tittel: 'Blues Brothers 2000', År: 1998, Land: 'USA', Sjanger: 'Komedie',  
 Alder: 11, Tid: 124, Pris: 135.00, Antall: 3},  
{FilmNr: 13, Tittel: 'Beatles: Help', År: 1965, Land: 'Storbritannia', Sjanger: 'Musikk',  
 Alder: 11, Tid: 144, Pris: 153.00, Antall: 0}  
]);

* 1. Henter Tittel, Sjanger og Pris for filmer som er produsert i 1988 eller seinere, sortert synkende på pris (=dyrest først).  
     db.film.find({År:{$gte:1988}}, { \_id:0, Tittel:1,Sjanger:1,Pris:1}). sort({Pris:-1}).pretty();
  2. Henter alle kolonner for filmer som ikke er til salgs (tomt på lager), sortert på alder og sjanger.  
     db.film.find({Antall: 0}).sort({Alder: 1},{Sjanger: 1}).pretty();

db.film.find({Antall: 0}, { \_id:0}).sort({Alder: 1},{Sjanger: 1}).pretty();

* 1. Finner antall filmer som er til salgs i hver sjanger, og summen av prisen

db.film.aggregate(

{ $group: {

\_id: "$Sjanger",

TotalPris: { $sum: '$Pris' }

}

});

* 1. Setter inn en ny rad. Finn på data selv.

db.film.**insert**(  
{FilmNr: 14, Tittel: 'De syv samuraier', År: 1955, Land: 'Japan', Sjanger: 'Action',  
 Alder: 15, Tid: 150, Pris: 149.00, Antall: 1});

db.film.**insert**(  
{FilmNr: 15, Tittel: 'Jorden rundt på 80 dager', År: 2021, Land: 'USA', Sjanger: 'Animasjon',  
 Alder: 7, Tid: 82, Pris: 120.00, Antall: 4});

* 1. Korrigerer tittelen på filmen "High Noon". Den heter egentlig "High Moon".  
     To versjoner:

db.film.update({Tittel: 'High Noon'},   
 { FilmNr: 5, Tittel: 'High Moon', År: 1952, Land: 'USA', Sjanger: 'Western',  
 Alder: 15, Tid: 85, Pris: 123.00, Antall: 2}); hele posten oppdateres  
db.film.update({Tittel: 'High Noon'}, { $set: {Tittel: 'High Moon'}}); bare aktuelt felt

* 1. Legger på 10% på prisen på alle Action filmer

db.film.updateMany(

{Sjanger: 'Action'},

[{ $set: { Pris: { $multiply: [ 1.1, "$Pris" ] } } }],

);  
db.film.update(

{Sjanger: 'Action'},

[{ $set: { Pris: { $multiply: [ 1.1, "$Pris" ] } } }],

{ multi: true }

);

* 1. Sletter filmen Anastasia.  
     db.film.remove({Tittel: 'Anastasia'});

1. **Mere MongoDB**
2. En kunde med navn **Kari Mo** med adresse **Esperantogata 22** i **Hamar** (postnummer **2300**) har kjøpt to filmer, nemlig **Apocalypse Now** og **High Moon**.  
   Opprett en dokumentsamling **faktura**  og sett inn post med følgende nøkler:  
    {**Fornavn, Etternavn, Adresse, PostNr, Poststed,   
    Filmer: [{FilmNr, Tittel, Pris, Antall}]** (array med ett element for hver kjøpt film)   
    **FakturaNr**, **Dato** (fakturadato)}  
   db.faktura.insert({  
    Fornavn: 'Kari', Etternavn: 'Mo', Adresse: 'Esperantogata 22', PostNr: 2300, Poststed: Hamar',  
    Filmer: [{FilmNr: 3,Tittel: 'Apocalypse Now', Pris: 123, Antall: 1},   
    { FilmNr: 5, Tittel: 'High Moon', Pris: 123, Antall: 2}],  
    FakturaNr: 1, Dato: Date()});

# Eksamen i 2DAN101-1 20H Databaser og nettverk

I dette eksamens-settet finner du først 3 oppgaver med **MongoDB**

1. Forklar strukturen i en **MongoDB**-data med utgangspunkt i data gitt under. Sammenlikne med en relasjonsdatabase.  
    [  
   {**"AnsNr": 1, "Fornavn": "Per", "Etternavn": "Hansen", "Telefon": [{"Tlf1":35950001, "Tlf2": 35951151}], "Gate": "Byvegen 12", "PostNr": 2300, "Poststed": "Hamar"},**{**"AnsNr": 2, "Fornavn": "Lise", "Etternavn": "Jensen", "Telefon": [{"Tlf1":35950002, "Tlf2": 91523344}], "Gate": "Liavegen 125", "PostNr": 2312, "Poststed": "Tangen"},**{**"AnsNr": 3, "Fornavn": "Anders", "Etternavn": "Lie", "Telefon": [{"Tlf1":35950003, "Tlf2": 35952249}], "Gate": "Stangevegen 17", "PostNr": 2310, "Poststed": "Stange"},**{**"AnsNr": 4, "Fornavn": "Johanne", "Etternavn": "Amundsen", "Telefon": [{"Tlf1":35950001, "Tlf2": 35951317, "Tlf3": 44827364}], "Gate": "Fossbakka 38", "Postnr": 2330, "Poststed": "Vallset"},**{**"AnsNr": 5, "Fornavn": "Arne", "Etternavn": "Lie", "Telefon": [{"Tlf1":35950004}],   
   "Gate": "Grønnegata 100", "Postnr": 2300, "Poststed": "Hamar"}**]  
   En **MongoDB**-database består av dokumentsamlinger ("collection", svarer til "tabell"). I eksemplet over er det én dokumentsamling som betår av 5 dokumenter.  
   Et dokument er et **JSON**-objekt: **[{ [{}],...},{...},...]**,og svarer til en rad i en tabell.Et **JSON**-objekt er en array **[...]** av objekter **{...}**. Disse objektene kan igjen inneholde arrayer med elementer eller nye objekter. Nesting av objekter/arrayer kan gjøres vilkårlig dypt.   
   Primærnøkkel genereres automatisk av systemet.
2. Åpne **MongoDB** i kommandovinduet. (Eller du kan bruke **Compass**.) Opprett en database **Eksamen2020** med en dokumentsamling **ansatt.** Legg inn dataene gitt foran.  
   >**use Eksamen2020;**  
   >**db.createCollection("ansatt");**>**db.ansatt.insert([...]);** der **[...]** er hele uttrykket for dokumentsamlingen gitt foran.
3. Skriv ut dataene på en oversiktig måte, og vis utskriftkommando.   
   Legg ved en skjermdump som viser noe av utskriften.  
   >**db.ansatt.find().pretty();**

De neste spørsmålene omhandler relasjonsdatabaser. Du står fritt til å benytte **MySQL Workbench** hvis du synes det er hensiktsmessig.

Oversett **MongoDB**-databasen foran til tabeller i en relasjonsdatabase slik at tabellene oppfyller **3NF**.  
Forklar skrittene du tar underveis.

**Utklipp fra div nettsider**

**1. Log Into MongoDB**

The following command can be used to log into the MongoDB database. Make sure that the user with credentials such as *username* and *password* exist in the database mentioned in place of dbname.

1

mongo -u <username> -p <password> --authenticationDatabase <dbname>

**2. Show All Databases**

Once logged in as a user with the appropriate role as userAdmin or userAdminAnyDatabase, the user can see all the databases using a command such as:

1

show dbs

**3. Select Database to Work With**

To start working with a particular database, the following command can be executed:

1

use databaseName

**4. Authenticate and Log Out From Database**

When switching to a different database using the use dbName command, the user is required to authenticate using a valid database user for that database. The following command can be used for authentication:

1

//

2

// Authenticate

3

//

4

db.auth("username", "password");

5

//

6

// Logout

7

//

8

db.logout()

**5. List Down Collections, Users, Roles, etc.**

The following commands can be used to check existing collections, users, etc.

1

//

2

// List down collections of the current database

3

//

4

show collections;

5

db.getCollectionNames();

6

//

7

// List down all the users of current database

8

//

9

show users;

10

db.getUsers();

11

//

12

// List down all the roles

13

//

14

show roles

**6. Create a Collection**

The following command can be used to create a collection. The details on this command can be found on [this page](https://docs.mongodb.com/manual/reference/method/db.createCollection/).

1

db.createCollection("collectionName");

**7. Insert a Document in a Collection**

Once a collection is created, the next step is to insert one or more documents. Following is a sample command for inserting a document in a collection.

1

//

2

// Insert single document

3

//

4

db.<collectionName>.insert({field1: "value", field2: "value"})

5

//

6

// Insert multiple documents

7

//

8

db.<collectionName>.insert([{field1: "value1"}, {field1: "value2"}])

9

db.<collectionName>.insertMany([{field1: "value1"}, {field1: "value2"}])

**8. Save or Update Document**

The save command can be used to either update an existing document or insert a new one depending on the document parameter passed to it. If the \_id passed matches an existing document, the document is updated. Otherwise, a new document is created. Internally, thesave method uses either the insert or the update command.

1

//

2

// Matching document will be updated; In case, no document matching the ID is found, a new document is created

3

//

4

db.<collectionName>.save({"\_id": new ObjectId("jhgsdjhgdsf"), field1: "value", field2: "value"});

**9. Display Collection Records**

The following commands can be used to retrieve collection records:

1

//

2

// Retrieve all records

3

//

4

db.<collectionName>.find();

5

//

6

// Retrieve limited number of records; Following command will print 10 results;

7

//

8

db.<collectionName>.find().limit(10);

9

//

10

// Retrieve records by id

11

//

12

db.<collectionName>.find({"\_id": ObjectId("someid")});

13

//

14

// Retrieve values of specific collection attributes by passing an object having

15

// attribute names assigned to 1 or 0 based on whether that attribute value needs

16

// to be included in the output or not, respectively.

17

//

18

db.<collectionName>.find({"\_id": ObjectId("someid")}, {field1: 1, field2: 1});

19

db.<collectionName>.find({"\_id": ObjectId("someid")}, {field1: 0}); // Exclude field1

20

//

21

// Collection count

22

//

23

db.<collectionName>.count();

**10. Administrative Commands**

Following are some of the administrative commands that can be helpful in finding collection details such as storage size, total size, and overall statistics.

1

//

2

// Get the collection statistics

3

//

4

db.<collectionName>.stats()

5

db.printCollectionStats()

6

//

7

// Latency statistics for read, writes operations including average time taken for reads, writes

8

// and related umber of operations performed

9

//

10

db.<collectionName>.latencyStats()

11

//

12

// Get collection size for data and indexes

13

//

14

db.<collectionName>.dataSize() // Size of the collection

15

db.<collectionName>.storageSize() // Total size of document stored in the collection

16

db.<collectionName>.totalSize() // Total size in bytes for both collection data and indexes

17

db.<collectionName>.totalIndexSize() // Total size of all indexes in the collection

And that's it! These commands will be of great help to all those just

#### 1. Create Database

In MongoDB use, DATABASE\_NAME is used to create a database. If this name database doesn’t exist, it will get created, and else it will return the existed one.

MongoDB command-Create Database

To check the current database now:



By default, MongoDB command comes with database name “test”. Suppose you inserted a document without specifying the database, it will automatically be stored in a “test” database.

#### 2. Drop Database

 MongoDB command -Drop Database

If the database is not specified, then it will delete the default database that is “test”.

#### 3. Create Collection

To create collection, the MongoDB command used is: **db.createCollection(name, options)**

Here, the name is the name of the collection & options is a document used to specify the configuration of the collection. Though “Options” parameter is optional, it’s good to provide it.

#### 4. Drop Collection



Drop Collection 2

#### 5. Insert Document

Insert() or save() method is used to insert data into any database collection.



Here “mycol” is the collection name. If the collection doesn’t exist, then MongoDB command will create the database collection and then it will get inserted.

#### 6. Query Document

Querying collection is done by find() method.

As find() method will show the findings in a non-structured way, to get the results in a structured pretty() method is used.

MongoDB command -Query Document

### Intermediate MongoDB Commands

#### 1. Limit()

This MongoDB command limits the no. of records need to use in MongoDB. The argument of this function accepts only number type. The argument is the number of the document that needs to be displayed.

Limit()

#### 2. Sort()

This is to the records of MongoDB. 1 & -1 are used to sort the documents. 1 is for ascending whereas -1 is for descending.

Sort()

#### 3. Indexing is the concept that helps MongoDB to scan documents inefficient way

Indexing

### Advanced Commands of  MongoDB

#### 1. Aggregate ()

This MongoDB command helps in processing the data, which returns the calculated result. This can group values from multiple documents together.

Aggregate

#### 2. Replication

[Replication in MongoDB](https://www.educba.com/replication-in-mongodb/) is achieved using a replication set. A replica set is a group of MongoDB processes that have the same dataset. Replica set provides:

1. High availability
2. Redundancy hence faults tolerant/disaster recovery.

In replica, one node is the primary node and rest others are the secondary node. All write operations remains with the primary node.

Let’s see; standalone MongoDB instance gets converted into a replica set.

Here are steps for that:

Close already running MongoDB server.

Now Start the MongoDB server by specifying — replSet option.

Syntax:

Syntax

#### 3. Create & restore Backup

To create the backup, mongodump command is used. The server’s entire data will be dumped into a dump directory(/bin/dump/). Options are there to limit the data.

Create & restore Backup

To restore the backup, a mongorestore command is used.

Create & restore Backup 2

#### 4. Monitor Deployment

To check the status of all your running processes/instances, a mongostat command is helpful. It tracks and returns the counter of database operations. These counters include inserts, updates, queries, deletes, and cursors. This MongoDB command is beneficial as it shows your status about low running memory, some performance issues, etc.

You need to go to your bin directory of MongoDB installation and run mongostat.

### Tips and Tricks to use MongoDB commands

* **Pre-allocate space:**When you know, your document is going to grow up to a certain size. This is an optimization technique in MongoDB. Insert a document and add a garbage field.
* Try fetching data in a single query.
* As MongoDB is by default case sensitive.

Example:

db.people.find({name: ‘Russell’}) &

db.people.find({name: ‘russell’}) are different.

While performing a search, its good habit of using regex. Like:

db.people.find({name: /russell/i})

* **Prefer Odd No. of Replica Sets:**An easy way to add redundancy and increase read performance is by using replica sets. Data is replicated between all nodes, and in case of primary node failure. Voting takes place between themselves, and the primary node is elected. Using the odd number of the replica will make voting easier in case of failure.
* **Secure MongoDB using a firewall:**As MongoDB itself doesn’t provide any authentication, it’s better to secure it with firewall and mapping it to the correct interface.
* **No joins:**As we know, joins are not supported by MongoDB. To retrieve data from more than two collections, one needs to write more than one query. And if the schema is not well organized, writing query may go hectic. This may result in the re-designing of the schema. It’s always better to spend some extra time to design a schema.

# **Database Commands**

All command documentation outlined below describes a command and its available parameters and provides a document template or prototype for each command. Some command documentation also includes the relevant [mongosh](https://docs.mongodb.com/mongodb-shell/" \l "mongodb-binary-bin.mongosh) helpers.

To run a command against the current database, use [db.runCommand()](https://docs.mongodb.com/manual/reference/method/db.runCommand/" \l "mongodb-method-db.runCommand):

|  |
| --- |
| db.runCommand( { <command> } ) |

To run an administrative command against the admin database, use [db.adminCommand()](https://docs.mongodb.com/manual/reference/method/db.adminCommand/" \l "mongodb-method-db.adminCommand):

|  |
| --- |
| db.adminCommand( { <command> } ) |

**NOTE**

For details on specific commands, including syntax and examples, click on the specific command to go to its reference page.

## Command Syntax

When you run a database command, you specify the command as a document to [db.runCommand()](https://docs.mongodb.com/manual/reference/method/db.runCommand/" \l "mongodb-method-db.runCommand). The document's key is the command to run, and the value is typically supplied as 1. The value does not affect the output of the command for example:

|  |
| --- |
| db.runCommand( { hello: 1 } ) |

## User Commands

### Aggregation Commands

| **Name** | **Description** |
| --- | --- |
| [aggregate](https://docs.mongodb.com/manual/reference/command/aggregate/#mongodb-dbcommand-dbcmd.aggregate) | Performs [aggregation tasks](https://docs.mongodb.com/manual/core/aggregation-pipeline/) such as group using the aggregation framework. |
| [count](https://docs.mongodb.com/manual/reference/command/count/#mongodb-dbcommand-dbcmd.count) | Counts the number of documents in a collection or a view. |
| [distinct](https://docs.mongodb.com/manual/reference/command/distinct/#mongodb-dbcommand-dbcmd.distinct) | Displays the distinct values found for a specified key in a collection or a view. |
| [mapReduce](https://docs.mongodb.com/manual/reference/command/mapReduce/#mongodb-dbcommand-dbcmd.mapReduce) | Performs [map-reduce](https://docs.mongodb.com/manual/core/map-reduce/) aggregation for large data sets. |

### Geospatial Commands

| **Name** | **Description** |
| --- | --- |
| [geoSearch](https://docs.mongodb.com/manual/reference/command/geoSearch/#mongodb-dbcommand-dbcmd.geoSearch) | Removed in MongoDB 5.0. Performs a geospatial query that uses MongoDB's [haystack index](https://docs.mongodb.com/manual/reference/glossary/#std-term-haystack-index) functionality. |

### Query and Write Operation Commands

| **Name** | **Description** |
| --- | --- |
| [delete](https://docs.mongodb.com/manual/reference/command/delete/#mongodb-dbcommand-dbcmd.delete) | Deletes one or more documents. |
| [find](https://docs.mongodb.com/manual/reference/command/find/#mongodb-dbcommand-dbcmd.find) | Selects documents in a collection or a view. |
| [findAndModify](https://docs.mongodb.com/manual/reference/command/findAndModify/#mongodb-dbcommand-dbcmd.findAndModify) | Returns and modifies a single document. |
| [getLastError](https://docs.mongodb.com/manual/reference/command/getLastError/#mongodb-dbcommand-dbcmd.getLastError) | Returns the success status of the last operation. |
| [getMore](https://docs.mongodb.com/manual/reference/command/getMore/#mongodb-dbcommand-dbcmd.getMore) | Returns batches of documents currently pointed to by the cursor. |
| [insert](https://docs.mongodb.com/manual/reference/command/insert/#mongodb-dbcommand-dbcmd.insert) | Inserts one or more documents. |
| [resetError](https://docs.mongodb.com/manual/reference/command/resetError/#mongodb-dbcommand-dbcmd.resetError) | Removed in MongoDB 5.0. Resets the last error status. |
| [update](https://docs.mongodb.com/manual/reference/command/update/#mongodb-dbcommand-dbcmd.update) | Updates one or more documents. |

### Query Plan Cache Commands

| **Name** | **Description** |
| --- | --- |
| [planCacheClear](https://docs.mongodb.com/manual/reference/command/planCacheClear/#mongodb-dbcommand-dbcmd.planCacheClear) | Removes cached query plan(s) for a collection. |
| [planCacheClearFilters](https://docs.mongodb.com/manual/reference/command/planCacheClearFilters/#mongodb-dbcommand-dbcmd.planCacheClearFilters) | Clears index filter(s) for a collection. |
| [planCacheListFilters](https://docs.mongodb.com/manual/reference/command/planCacheListFilters/#mongodb-dbcommand-dbcmd.planCacheListFilters) | Lists the index filters for a collection. |
| [planCacheSetFilter](https://docs.mongodb.com/manual/reference/command/planCacheSetFilter/#mongodb-dbcommand-dbcmd.planCacheSetFilter) | Sets an index filter for a collection. |

## Database Operations

### Authentication Commands

| **Name** | **Description** |
| --- | --- |
| [authenticate](https://docs.mongodb.com/manual/reference/command/authenticate/#mongodb-dbcommand-dbcmd.authenticate) | Starts an authenticated session using a username and password. |
| [getnonce](https://docs.mongodb.com/manual/reference/command/getnonce/#mongodb-dbcommand-dbcmd.getnonce) | This is an internal command to generate a one-time password for authentication. |
| [logout](https://docs.mongodb.com/manual/reference/command/logout/#mongodb-dbcommand-dbcmd.logout) | Terminates the current authenticated session. |

### User Management Commands

| **Name** | **Description** |
| --- | --- |
| [createUser](https://docs.mongodb.com/manual/reference/command/createUser/#mongodb-dbcommand-dbcmd.createUser) | Creates a new user. |
| [dropAllUsersFromDatabase](https://docs.mongodb.com/manual/reference/command/dropAllUsersFromDatabase/#mongodb-dbcommand-dbcmd.dropAllUsersFromDatabase) | Deletes all users associated with a database. |
| [dropUser](https://docs.mongodb.com/manual/reference/command/dropUser/#mongodb-dbcommand-dbcmd.dropUser) | Removes a single user. |
| [grantRolesToUser](https://docs.mongodb.com/manual/reference/command/grantRolesToUser/#mongodb-dbcommand-dbcmd.grantRolesToUser) | Grants a role and its privileges to a user. |
| [revokeRolesFromUser](https://docs.mongodb.com/manual/reference/command/revokeRolesFromUser/#mongodb-dbcommand-dbcmd.revokeRolesFromUser) | Removes a role from a user. |
| [updateUser](https://docs.mongodb.com/manual/reference/command/updateUser/#mongodb-dbcommand-dbcmd.updateUser) | Updates a user's data. |
| [usersInfo](https://docs.mongodb.com/manual/reference/command/usersInfo/#mongodb-dbcommand-dbcmd.usersInfo) | Returns information about the specified users. |

### Role Management Commands

| **Name** | **Description** |
| --- | --- |
| [createRole](https://docs.mongodb.com/manual/reference/command/createRole/#mongodb-dbcommand-dbcmd.createRole) | Creates a role and specifies its privileges. |
| [dropRole](https://docs.mongodb.com/manual/reference/command/dropRole/#mongodb-dbcommand-dbcmd.dropRole) | Deletes the user-defined role. |
| [dropAllRolesFromDatabase](https://docs.mongodb.com/manual/reference/command/dropAllRolesFromDatabase/#mongodb-dbcommand-dbcmd.dropAllRolesFromDatabase) | Deletes all user-defined roles from a database. |
| [grantPrivilegesToRole](https://docs.mongodb.com/manual/reference/command/grantPrivilegesToRole/#mongodb-dbcommand-dbcmd.grantPrivilegesToRole) | Assigns privileges to a user-defined role. |
| [grantRolesToRole](https://docs.mongodb.com/manual/reference/command/grantRolesToRole/#mongodb-dbcommand-dbcmd.grantRolesToRole) | Specifies roles from which a user-defined role inherits privileges. |
| [invalidateUserCache](https://docs.mongodb.com/manual/reference/command/invalidateUserCache/#mongodb-dbcommand-dbcmd.invalidateUserCache) | Flushes the in-memory cache of user information, including credentials and roles. |
| [revokePrivilegesFromRole](https://docs.mongodb.com/manual/reference/command/revokePrivilegesFromRole/#mongodb-dbcommand-dbcmd.revokePrivilegesFromRole) | Removes the specified privileges from a user-defined role. |
| [revokeRolesFromRole](https://docs.mongodb.com/manual/reference/command/revokeRolesFromRole/#mongodb-dbcommand-dbcmd.revokeRolesFromRole) | Removes specified inherited roles from a user-defined role. |
| [rolesInfo](https://docs.mongodb.com/manual/reference/command/rolesInfo/#mongodb-dbcommand-dbcmd.rolesInfo) | Returns information for the specified role or roles. |
| [updateRole](https://docs.mongodb.com/manual/reference/command/updateRole/#mongodb-dbcommand-dbcmd.updateRole) | Updates a user-defined role. |

### Replication Commands

| **Name** | **Description** |
| --- | --- |
| [applyOps](https://docs.mongodb.com/manual/reference/command/applyOps/#mongodb-dbcommand-dbcmd.applyOps) | Internal command that applies [oplog](https://docs.mongodb.com/manual/reference/glossary/" \l "std-term-oplog) entries to the current data set. |
| [hello](https://docs.mongodb.com/manual/reference/command/hello/#mongodb-dbcommand-dbcmd.hello) | Displays information about this member's role in the replica set, including whether it is the primary. |
| [replSetAbortPrimaryCatchUp](https://docs.mongodb.com/manual/reference/command/replSetAbortPrimaryCatchUp/#mongodb-dbcommand-dbcmd.replSetAbortPrimaryCatchUp) | Forces the elected [primary](https://docs.mongodb.com/manual/reference/glossary/#std-term-primary) to abort sync (catch up) then complete the transition to primary. |
| [replSetFreeze](https://docs.mongodb.com/manual/reference/command/replSetFreeze/#mongodb-dbcommand-dbcmd.replSetFreeze) | Prevents the current member from seeking election as [primary](https://docs.mongodb.com/manual/reference/glossary/#std-term-primary) for a period of time. |
| [replSetGetConfig](https://docs.mongodb.com/manual/reference/command/replSetGetConfig/#mongodb-dbcommand-dbcmd.replSetGetConfig) | Returns the replica set's configuration object. |
| [replSetGetStatus](https://docs.mongodb.com/manual/reference/command/replSetGetStatus/#mongodb-dbcommand-dbcmd.replSetGetStatus) | Returns a document that reports on the status of the replica set. |
| [replSetInitiate](https://docs.mongodb.com/manual/reference/command/replSetInitiate/#mongodb-dbcommand-dbcmd.replSetInitiate) | Initializes a new replica set. |
| [replSetMaintenance](https://docs.mongodb.com/manual/reference/command/replSetMaintenance/#mongodb-dbcommand-dbcmd.replSetMaintenance) | Enables or disables a maintenance mode, which puts a [secondary](https://docs.mongodb.com/manual/reference/glossary/#std-term-secondary) node in a RECOVERING state. |
| [replSetReconfig](https://docs.mongodb.com/manual/reference/command/replSetReconfig/#mongodb-dbcommand-dbcmd.replSetReconfig) | Applies a new configuration to an existing replica set. |
| [replSetResizeOplog](https://docs.mongodb.com/manual/reference/command/replSetResizeOplog/#mongodb-dbcommand-dbcmd.replSetResizeOplog) | Dynamically resizes the oplog for a replica set member. Available for WiredTiger storage engine only. |
| [replSetStepDown](https://docs.mongodb.com/manual/reference/command/replSetStepDown/#mongodb-dbcommand-dbcmd.replSetStepDown) | Forces the current [primary](https://docs.mongodb.com/manual/reference/glossary/#std-term-primary) to step down and become a [secondary](https://docs.mongodb.com/manual/reference/glossary/#std-term-secondary), forcing an election. |
| [replSetSyncFrom](https://docs.mongodb.com/manual/reference/command/replSetSyncFrom/#mongodb-dbcommand-dbcmd.replSetSyncFrom) | Explicitly override the default logic for selecting a member to replicate from. |

**TIP**

**See also:**

[Replication](https://docs.mongodb.com/manual/replication/) for more information regarding replication.

### Sharding Commands

| **Name** | **Description** |
| --- | --- |
| [abortReshardCollection](https://docs.mongodb.com/manual/reference/command/abortReshardCollection/#mongodb-dbcommand-dbcmd.abortReshardCollection) | Aborts a [resharding operation](https://docs.mongodb.com/manual/core/sharding-reshard-a-collection/" \l "std-label-sharding-resharding).  New in version 5.0. |
| [addShard](https://docs.mongodb.com/manual/reference/command/addShard/#mongodb-dbcommand-dbcmd.addShard) | Adds a [shard](https://docs.mongodb.com/manual/reference/glossary/#std-term-shard) to a [sharded cluster](https://docs.mongodb.com/manual/reference/glossary/#std-term-sharded-cluster). |
| [addShardToZone](https://docs.mongodb.com/manual/reference/command/addShardToZone/#mongodb-dbcommand-dbcmd.addShardToZone) | Associates a shard with a [zone](https://docs.mongodb.com/manual/reference/glossary/#std-term-zone). Supports configuring [zones](https://docs.mongodb.com/manual/core/zone-sharding/" \l "std-label-zone-sharding) in sharded clusters. |
| [balancerCollectionStatus](https://docs.mongodb.com/manual/reference/command/balancerCollectionStatus/#mongodb-dbcommand-dbcmd.balancerCollectionStatus) | Returns information on whether the chunks of a sharded collection are balanced.  New in version 4.4. |
| [balancerStart](https://docs.mongodb.com/manual/reference/command/balancerStart/#mongodb-dbcommand-dbcmd.balancerStart) | Starts a balancer thread. |
| [balancerStatus](https://docs.mongodb.com/manual/reference/command/balancerStatus/#mongodb-dbcommand-dbcmd.balancerStatus) | Returns information on the balancer status. |
| [balancerStop](https://docs.mongodb.com/manual/reference/command/balancerStop/#mongodb-dbcommand-dbcmd.balancerStop) | Stops the balancer thread. |
| [checkShardingIndex](https://docs.mongodb.com/manual/reference/command/checkShardingIndex/#mongodb-dbcommand-dbcmd.checkShardingIndex) | Internal command that validates index on shard key. |
| [clearJumboFlag](https://docs.mongodb.com/manual/reference/command/clearJumboFlag/#mongodb-dbcommand-dbcmd.clearJumboFlag) | Clears the jumbo flag for a chunk. |
| [cleanupOrphaned](https://docs.mongodb.com/manual/reference/command/cleanupOrphaned/#mongodb-dbcommand-dbcmd.cleanupOrphaned) | Removes orphaned data with shard key values outside of the ranges of the chunks owned by a shard. |
| [cleanupReshardCollection](https://docs.mongodb.com/manual/reference/command/cleanupReshardCollection/#mongodb-dbcommand-dbcmd.cleanupReshardCollection) | Cleans up a failed [resharding operation](https://docs.mongodb.com/manual/core/sharding-reshard-a-collection/" \l "std-label-sharding-resharding).  New in version 5.0. |
| [commitReshardCollection](https://docs.mongodb.com/manual/reference/command/commitReshardCollection/#mongodb-dbcommand-dbcmd.commitReshardCollection) | Forces a [resharding operation](https://docs.mongodb.com/manual/core/sharding-reshard-a-collection/" \l "std-label-sharding-resharding) to block writes and complete.  New in version 5.0. |
| [enableSharding](https://docs.mongodb.com/manual/reference/command/enableSharding/#mongodb-dbcommand-dbcmd.enableSharding) | Enables sharding on a specific database. |
| [flushRouterConfig](https://docs.mongodb.com/manual/reference/command/flushRouterConfig/#mongodb-dbcommand-dbcmd.flushRouterConfig) | Forces a [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "mongodb-binary-bin.mongod)/[mongos](https://docs.mongodb.com/manual/reference/program/mongos/#mongodb-binary-bin.mongos) instance to update its cached routing metadata. |
| [getShardMap](https://docs.mongodb.com/manual/reference/command/getShardMap/#mongodb-dbcommand-dbcmd.getShardMap) | Internal command that reports on the state of a sharded cluster. |
| [getShardVersion](https://docs.mongodb.com/manual/reference/command/getShardVersion/#mongodb-dbcommand-dbcmd.getShardVersion) | Internal command that returns the [config server](https://docs.mongodb.com/manual/reference/glossary/#std-term-config-database) version. |
| [isdbgrid](https://docs.mongodb.com/manual/reference/command/isdbgrid/#mongodb-dbcommand-dbcmd.isdbgrid) | Verifies that a process is a [mongos](https://docs.mongodb.com/manual/reference/program/mongos/#mongodb-binary-bin.mongos). |
| [listShards](https://docs.mongodb.com/manual/reference/command/listShards/#mongodb-dbcommand-dbcmd.listShards) | Returns a list of configured shards. |
| [medianKey](https://docs.mongodb.com/manual/reference/command/medianKey/#mongodb-dbcommand-dbcmd.medianKey) | Deprecated internal command. See [splitVector](https://docs.mongodb.com/manual/reference/command/splitVector/" \l "mongodb-dbcommand-dbcmd.splitVector). |
| [moveChunk](https://docs.mongodb.com/manual/reference/command/moveChunk/#mongodb-dbcommand-dbcmd.moveChunk) | Internal command that migrates chunks between shards. |
| [movePrimary](https://docs.mongodb.com/manual/reference/command/movePrimary/#mongodb-dbcommand-dbcmd.movePrimary) | Reassigns the [primary shard](https://docs.mongodb.com/manual/reference/glossary/#std-term-primary-shard) when removing a shard from a sharded cluster. |
| [mergeChunks](https://docs.mongodb.com/manual/reference/command/mergeChunks/#mongodb-dbcommand-dbcmd.mergeChunks) | Provides the ability to combine chunks on a single shard. |
| [refineCollectionShardKey](https://docs.mongodb.com/manual/reference/command/refineCollectionShardKey/#mongodb-dbcommand-dbcmd.refineCollectionShardKey) | Refines a collection's shard key by adding a suffix to the existing key.  New in version 4.4. |
| [removeShard](https://docs.mongodb.com/manual/reference/command/removeShard/#mongodb-dbcommand-dbcmd.removeShard) | Starts the process of removing a shard from a sharded cluster. |
| [removeShardFromZone](https://docs.mongodb.com/manual/reference/command/removeShardFromZone/#mongodb-dbcommand-dbcmd.removeShardFromZone) | Removes the association between a shard and a [zone](https://docs.mongodb.com/manual/reference/glossary/#std-term-zone). Supports configuring [zones](https://docs.mongodb.com/manual/core/zone-sharding/" \l "std-label-zone-sharding) in sharded clusters. |
| [reshardCollection](https://docs.mongodb.com/manual/reference/command/reshardCollection/#mongodb-dbcommand-dbcmd.reshardCollection) | Initiates a [resharding operation](https://docs.mongodb.com/manual/core/sharding-reshard-a-collection/" \l "std-label-sharding-resharding) to change the shard key for a collection, changing the distribution of your data.  New in version 5.0. |
| [setShardVersion](https://docs.mongodb.com/manual/reference/command/setShardVersion/#mongodb-dbcommand-dbcmd.setShardVersion) | Internal command to sets the [config server](https://docs.mongodb.com/manual/reference/glossary/#std-term-config-database) version. |
| [shardCollection](https://docs.mongodb.com/manual/reference/command/shardCollection/#mongodb-dbcommand-dbcmd.shardCollection) | Enables the sharding functionality for a collection, allowing the collection to be sharded. |
| [shardingState](https://docs.mongodb.com/manual/reference/command/shardingState/#mongodb-dbcommand-dbcmd.shardingState) | Reports whether the [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "mongodb-binary-bin.mongod) is a member of a sharded cluster. |
| [split](https://docs.mongodb.com/manual/reference/command/split/#mongodb-dbcommand-dbcmd.split) | Creates a new [chunk](https://docs.mongodb.com/manual/reference/glossary/" \l "std-term-chunk). |
| [splitChunk](https://docs.mongodb.com/manual/reference/command/splitChunk/#mongodb-dbcommand-dbcmd.splitChunk) | Internal command to split chunk. Instead use the methods [sh.splitFind()](https://docs.mongodb.com/manual/reference/method/sh.splitFind/" \l "mongodb-method-sh.splitFind) and [sh.splitAt()](https://docs.mongodb.com/manual/reference/method/sh.splitAt/" \l "mongodb-method-sh.splitAt). |
| [splitVector](https://docs.mongodb.com/manual/reference/command/splitVector/#mongodb-dbcommand-dbcmd.splitVector) | Internal command that determines split points. |
| [unsetSharding](https://docs.mongodb.com/manual/reference/command/unsetSharding/#mongodb-dbcommand-dbcmd.unsetSharding) | Removed in MongoDB 5.0. Internal command that affects connections between instances in a MongoDB deployment. |
| [updateZoneKeyRange](https://docs.mongodb.com/manual/reference/command/updateZoneKeyRange/#mongodb-dbcommand-dbcmd.updateZoneKeyRange) | Adds or removes the association between a range of sharded data and a [zone](https://docs.mongodb.com/manual/reference/glossary/#std-term-zone). Supports configuring [zones](https://docs.mongodb.com/manual/core/zone-sharding/" \l "std-label-zone-sharding) in sharded clusters. |

**TIP**

**See also:**

[Sharding](https://docs.mongodb.com/manual/sharding/) for more information about MongoDB's sharding functionality.

### Session Commands

| **Command** | **Description** |
| --- | --- |
| [abortTransaction](https://docs.mongodb.com/manual/reference/command/abortTransaction/#mongodb-dbcommand-dbcmd.abortTransaction) | Abort transaction.  New in version 4.0. |
| [commitTransaction](https://docs.mongodb.com/manual/reference/command/commitTransaction/#mongodb-dbcommand-dbcmd.commitTransaction) | Commit transaction.  New in version 4.0. |
| [endSessions](https://docs.mongodb.com/manual/reference/command/endSessions/#mongodb-dbcommand-dbcmd.endSessions) | Expire sessions before the sessions' timeout period.  New in version 3.6. |
| [killAllSessions](https://docs.mongodb.com/manual/reference/command/killAllSessions/#mongodb-dbcommand-dbcmd.killAllSessions) | Kill all sessions.  New in version 3.6. |
| [killAllSessionsByPattern](https://docs.mongodb.com/manual/reference/command/killAllSessionsByPattern/#mongodb-dbcommand-dbcmd.killAllSessionsByPattern) | Kill all sessions that match the specified pattern  New in version 3.6. |
| [killSessions](https://docs.mongodb.com/manual/reference/command/killSessions/#mongodb-dbcommand-dbcmd.killSessions) | Kill specified sessions.  New in version 3.6. |
| [refreshSessions](https://docs.mongodb.com/manual/reference/command/refreshSessions/#mongodb-dbcommand-dbcmd.refreshSessions) | Refresh idle sessions.  New in version 3.6. |
| [startSession](https://docs.mongodb.com/manual/reference/command/startSession/#mongodb-dbcommand-dbcmd.startSession) | Starts a new session.  New in version 3.6. |

### Administration Commands

| **Name** | **Description** |
| --- | --- |
| [cloneCollectionAsCapped](https://docs.mongodb.com/manual/reference/command/cloneCollectionAsCapped/#mongodb-dbcommand-dbcmd.cloneCollectionAsCapped) | Copies a non-capped collection as a new [capped collection](https://docs.mongodb.com/manual/reference/glossary/#std-term-capped-collection). |
| [collMod](https://docs.mongodb.com/manual/reference/command/collMod/#mongodb-dbcommand-dbcmd.collMod) | Add options to a collection or modify a view definition. |
| [compact](https://docs.mongodb.com/manual/reference/command/compact/#mongodb-dbcommand-dbcmd.compact) | Defragments a collection and rebuilds the indexes. |
| [connPoolSync](https://docs.mongodb.com/manual/reference/command/connPoolSync/#mongodb-dbcommand-dbcmd.connPoolSync) | Internal command to flush connection pool. |
| [convertToCapped](https://docs.mongodb.com/manual/reference/command/convertToCapped/#mongodb-dbcommand-dbcmd.convertToCapped) | Converts a non-capped collection to a capped collection. |
| [create](https://docs.mongodb.com/manual/reference/command/create/#mongodb-dbcommand-dbcmd.create) | Creates a collection or a view. |
| [createIndexes](https://docs.mongodb.com/manual/reference/command/createIndexes/#mongodb-dbcommand-dbcmd.createIndexes) | Builds one or more indexes for a collection. |
| [currentOp](https://docs.mongodb.com/manual/reference/command/currentOp/#mongodb-dbcommand-dbcmd.currentOp) | Returns a document that contains information on in-progress operations for the database instance. |
| [drop](https://docs.mongodb.com/manual/reference/command/drop/#mongodb-dbcommand-dbcmd.drop) | Removes the specified collection from the database. |
| [dropDatabase](https://docs.mongodb.com/manual/reference/command/dropDatabase/#mongodb-dbcommand-dbcmd.dropDatabase) | Removes the current database. |
| [dropConnections](https://docs.mongodb.com/manual/reference/command/dropConnections/#mongodb-dbcommand-dbcmd.dropConnections) | Drops outgoing connections to the specified list of hosts. |
| [dropIndexes](https://docs.mongodb.com/manual/reference/command/dropIndexes/#mongodb-dbcommand-dbcmd.dropIndexes) | Removes indexes from a collection. |
| [filemd5](https://docs.mongodb.com/manual/reference/command/filemd5/#mongodb-dbcommand-dbcmd.filemd5) | Returns the [md5](https://docs.mongodb.com/manual/reference/glossary/#std-term-md5) hash for files stored using [GridFS](https://docs.mongodb.com/manual/reference/glossary/" \l "std-term-GridFS). |
| [fsync](https://docs.mongodb.com/manual/reference/command/fsync/#mongodb-dbcommand-dbcmd.fsync) | Flushes pending writes to the storage layer and locks the database to allow backups. |
| [fsyncUnlock](https://docs.mongodb.com/manual/reference/command/fsyncUnlock/#mongodb-dbcommand-dbcmd.fsyncUnlock) | Unlocks one fsync lock. |
| [getDefaultRWConcern](https://docs.mongodb.com/manual/reference/command/getDefaultRWConcern/#mongodb-dbcommand-dbcmd.getDefaultRWConcern) | Retrieves the global default read and write concern options for the deployment.  New in version 4.4. |
| [getParameter](https://docs.mongodb.com/manual/reference/command/getParameter/#mongodb-dbcommand-dbcmd.getParameter) | Retrieves configuration options. |
| [killCursors](https://docs.mongodb.com/manual/reference/command/killCursors/#mongodb-dbcommand-dbcmd.killCursors) | Kills the specified cursors for a collection. |
| [killOp](https://docs.mongodb.com/manual/reference/command/killOp/#mongodb-dbcommand-dbcmd.killOp) | Terminates an operation as specified by the operation ID. |
| [listCollections](https://docs.mongodb.com/manual/reference/command/listCollections/#mongodb-dbcommand-dbcmd.listCollections) | Returns a list of collections in the current database. |
| [listDatabases](https://docs.mongodb.com/manual/reference/command/listDatabases/#mongodb-dbcommand-dbcmd.listDatabases) | Returns a document that lists all databases and returns basic database statistics. |
| [listIndexes](https://docs.mongodb.com/manual/reference/command/listIndexes/#mongodb-dbcommand-dbcmd.listIndexes) | Lists all indexes for a collection. |
| [logRotate](https://docs.mongodb.com/manual/reference/command/logRotate/#mongodb-dbcommand-dbcmd.logRotate) | Rotates the MongoDB logs to prevent a single file from taking too much space. |
| [reIndex](https://docs.mongodb.com/manual/reference/command/reIndex/#mongodb-dbcommand-dbcmd.reIndex) | Rebuilds all indexes on a collection. |
| [renameCollection](https://docs.mongodb.com/manual/reference/command/renameCollection/#mongodb-dbcommand-dbcmd.renameCollection) | Changes the name of an existing collection. |
| [rotateCertificates](https://docs.mongodb.com/manual/reference/command/rotateCertificates/#mongodb-dbcommand-dbcmd.rotateCertificates) | Performs online TLS certificate rotation. |
| [setFeatureCompatibilityVersion](https://docs.mongodb.com/manual/reference/command/setFeatureCompatibilityVersion/#mongodb-dbcommand-dbcmd.setFeatureCompatibilityVersion) | Enables or disables features that persist data that are backwards-incompatible. |
| [setIndexCommitQuorum](https://docs.mongodb.com/manual/reference/command/setIndexCommitQuorum/#mongodb-dbcommand-dbcmd.setIndexCommitQuorum) | Changes the minimum number of data-bearing members (i.e commit quorum), including the primary, that must vote to commit an in-progress [index build](https://docs.mongodb.com/manual/core/index-creation/#std-label-index-operations-replicated-build) before the primary marks those indexes as ready. |
| [setParameter](https://docs.mongodb.com/manual/reference/command/setParameter/#mongodb-dbcommand-dbcmd.setParameter) | Modifies configuration options. |
| [setDefaultRWConcern](https://docs.mongodb.com/manual/reference/command/setDefaultRWConcern/#mongodb-dbcommand-dbcmd.setDefaultRWConcern) | Sets the global default read and write concern options for the deployment.  New in version 4.4. |
| [shutdown](https://docs.mongodb.com/manual/reference/command/shutdown/#mongodb-dbcommand-dbcmd.shutdown) | Shuts down the [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "mongodb-binary-bin.mongod) or [mongos](https://docs.mongodb.com/manual/reference/program/mongos/#mongodb-binary-bin.mongos) process. |

### Diagnostic Commands

| **Name** | **Description** |
| --- | --- |
| [availableQueryOptions](https://docs.mongodb.com/manual/reference/command/availableQueryOptions/#mongodb-dbcommand-dbcmd.availableQueryOptions) | Internal command that reports on the capabilities of the current MongoDB instance. |
| [buildInfo](https://docs.mongodb.com/manual/reference/command/buildInfo/#mongodb-dbcommand-dbcmd.buildInfo) | Displays statistics about the MongoDB build. |
| [collStats](https://docs.mongodb.com/manual/reference/command/collStats/#mongodb-dbcommand-dbcmd.collStats) | Reports storage utilization statics for a specified collection. |
| [connPoolStats](https://docs.mongodb.com/manual/reference/command/connPoolStats/#mongodb-dbcommand-dbcmd.connPoolStats) | Reports statistics on the outgoing connections from this MongoDB instance to other MongoDB instances in the deployment. |
| [connectionStatus](https://docs.mongodb.com/manual/reference/command/connectionStatus/#mongodb-dbcommand-dbcmd.connectionStatus) | Reports the authentication state for the current connection. |
| [cursorInfo](https://docs.mongodb.com/manual/reference/command/cursorInfo/#mongodb-dbcommand-dbcmd.cursorInfo) | Removed in MongoDB 3.2. Replaced with [metrics.cursor](https://docs.mongodb.com/manual/reference/command/serverStatus/" \l "mongodb-serverstatus-serverstatus.metrics.cursor). |
| [dataSize](https://docs.mongodb.com/manual/reference/command/dataSize/#mongodb-dbcommand-dbcmd.dataSize) | Returns the data size for a range of data. For internal use. |
| [dbHash](https://docs.mongodb.com/manual/reference/command/dbHash/#mongodb-dbcommand-dbcmd.dbHash) | Returns hash value a database and its collections. |
| [dbStats](https://docs.mongodb.com/manual/reference/command/dbStats/#mongodb-dbcommand-dbcmd.dbStats) | Reports storage utilization statistics for the specified database. |
| [driverOIDTest](https://docs.mongodb.com/manual/reference/command/driverOIDTest/#mongodb-dbcommand-dbcmd.driverOIDTest) | Internal command that converts an ObjectId to a string to support tests. |
| [explain](https://docs.mongodb.com/manual/reference/command/explain/#mongodb-dbcommand-dbcmd.explain) | Returns information on the execution of various operations. |
| [features](https://docs.mongodb.com/manual/reference/command/features/#mongodb-dbcommand-dbcmd.features) | Reports on features available in the current MongoDB instance. |
| [getCmdLineOpts](https://docs.mongodb.com/manual/reference/command/getCmdLineOpts/#mongodb-dbcommand-dbcmd.getCmdLineOpts) | Returns a document with the run-time arguments to the MongoDB instance and their parsed options. |
| [getLog](https://docs.mongodb.com/manual/reference/command/getLog/#mongodb-dbcommand-dbcmd.getLog) | Returns recent log messages. |
| [hostInfo](https://docs.mongodb.com/manual/reference/command/hostInfo/#mongodb-dbcommand-dbcmd.hostInfo) | Returns data that reflects the underlying host system. |
| [\_isSelf](https://docs.mongodb.com/manual/reference/command/isSelf/#mongodb-dbcommand-dbcmd._isSelf) | Internal command to support testing. |
| [listCommands](https://docs.mongodb.com/manual/reference/command/listCommands/#mongodb-dbcommand-dbcmd.listCommands) | Lists all database commands provided by the current [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "mongodb-binary-bin.mongod) instance. |
| [lockInfo](https://docs.mongodb.com/manual/reference/command/lockInfo/#mongodb-dbcommand-dbcmd.lockInfo) | Internal command that returns information on locks that are currently being held or pending. Only available for [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "mongodb-binary-bin.mongod) instances. |
| [netstat](https://docs.mongodb.com/manual/reference/command/netstat/#mongodb-dbcommand-dbcmd.netstat) | Internal command that reports on intra-deployment connectivity. Only available for [mongos](https://docs.mongodb.com/manual/reference/program/mongos/#mongodb-binary-bin.mongos) instances. |
| [ping](https://docs.mongodb.com/manual/reference/command/ping/#mongodb-dbcommand-dbcmd.ping) | Internal command that tests intra-deployment connectivity. |
| [profile](https://docs.mongodb.com/manual/reference/command/profile/#mongodb-dbcommand-dbcmd.profile) | Interface for the [database profiler](https://docs.mongodb.com/manual/reference/database-profiler/#std-label-profiler). |
| [serverStatus](https://docs.mongodb.com/manual/reference/command/serverStatus/#mongodb-dbcommand-dbcmd.serverStatus) | Returns a collection metrics on instance-wide resource utilization and status. |
| [shardConnPoolStats](https://docs.mongodb.com/manual/reference/command/shardConnPoolStats/#mongodb-dbcommand-dbcmd.shardConnPoolStats) | Removed in MongoDB 5.0. Use [connPoolStats](https://docs.mongodb.com/manual/reference/command/connPoolStats/" \l "mongodb-dbcommand-dbcmd.connPoolStats) instead. |
| [top](https://docs.mongodb.com/manual/reference/command/top/#mongodb-dbcommand-dbcmd.top) | Returns raw usage statistics for each database in the [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "mongodb-binary-bin.mongod) instance. |
| [validate](https://docs.mongodb.com/manual/reference/command/validate/#mongodb-dbcommand-dbcmd.validate) | Internal command that scans for a collection's data and indexes for correctness. |
| [whatsmyuri](https://docs.mongodb.com/manual/reference/command/whatsmyuri/#mongodb-dbcommand-dbcmd.whatsmyuri) | Internal command that returns information on the current client. |

### Free Monitoring Commands

| **Name** | **Description** |
| --- | --- |
| [setFreeMonitoring](https://docs.mongodb.com/manual/reference/command/setFreeMonitoring/#mongodb-dbcommand-dbcmd.setFreeMonitoring) | Enables/disables free monitoring during runtime. |

## Auditing Commands

| **Name** | **Description** |
| --- | --- |
| [logApplicationMessage](https://docs.mongodb.com/manual/reference/command/logApplicationMessage/#mongodb-dbcommand-dbcmd.logApplicationMessage) | Posts a custom message to the audit log. |