



MongoDB Utilities For Exporting and importing data from and into MongoDB



Introduction

- Mongo DB utilities are used for Data Management and Monitoring.
- These utilities are not part of MongoDB shell.



MongoDB Utilities

mongoexport

mongoimport

mongodump

mongorestore

mongostat

A decorative sidebar on the left side of the slide, featuring a grid of blue and white icons. The icons include a television, a camera, a lightbulb, a speech bubble, a hand, a smartphone, a shopping cart, a Twitter bird, and a large letter 't'.

Instructions for using utilites

- Initiate the MongoDB server using this command: *mongod*
- Then, in a new terminal/command prompt (cmd) in *administrative mode*
- **I**nitiate the MongoDB client system using the appropriate mongoddb utility command



What does export of Data mean?

- Data export is either an automatic or semi-automated task that changes a file from one format to another.
- It involves "interpreting" a set of data from one format used in a particular application into another.

- Exports MongoDB collection data in JSON or CSV(Comma separated values) format
- Strict Mode Extended JSON format is used

A decorative header featuring a grid of various icons in shades of blue and white, including a wrench, car, sun, shopping cart, briefcase, smartphone, family, apple, Wi-Fi, headphones, and a book.

What mongoexport does?

Collection

Csv/json/text
file/json array

Example 1

```
mongoexport --db myDb --collection persons --out  
persons.json
```

Where:

- `--db` specifies the database name
- `--collection` specifies the collection name
- `--out` specifies the path of the output file
- By default, the `mongoexport` command writes the data using one JSON document for every Mongo document. Using the optional `--jsonArray` parameter modifies the output of the `mongoexport` command as a single JSON array
- `--pretty` is again an optional parameter that outputs the collection in a pretty-printed JSON format



Example 2: Exporting from Remote Machine

```
mongoexport --host ec2-34-216-244-19.us-  
west-2.compute.amazonaws.com --username admin --  
password password --authenticationDatabase admin  
--db myDb --collection persons --out persons.json
```

Mongoexport syntax

```
mongoexport --host <hostname> --username  
<username> --password <password> --  
authenticationDatabase <authDb> --db  
<database> --collection <collection> --out  
<filename>
```

Notes

--jsonArray Option will export data
as Array of JSON objects



Mongo database Export syntax

```
1 | > mongoexport --host <host_name> --username <user_name> --password <password> --db <database_r
```

Where:

- `--host` is an optional parameter that specifies the remote server Mongo database instance
- `--username` and `--password` are the optional parameters that specify the authentication details of a user
- `--db` specifies the database name
- `--collection` specifies the collection name
- `--out` specifies the path of the output file. If this is *not* specified, the result is displayed on the console

Exporting into different types of files

(REMOTE)

```
mongoexport --host <hostname> --username <username> --password <password> --  
authenticationDatabase <authDb> --db <database> --collection <collection> --out  
<filename>
```

(LOCAL)

```
mongoexport --db <database> --collection <collection> --out <filename>
```

(LOCAL jsonArray)

```
mongoexport --db <database> --collection <collection> --out <filename> --jsonArray
```

(LOCAL CSV)

```
mongoexport --db <database> --collection <collection> --out <filename> --type=csv
```

(LOCAL with Query)

```
mongoexport --db <database> --collection <collection> --out <filename> --query  
'{<query>}'
```

- 
- Hands-on on mongoexport



(Export to json)

```
mongoexport --db person --collection personData --out  
per.json
```

(Export to json Array)

```
mongoexport --db person --collection personData --out  
per.json --jsonArray
```



(Export to csv file)

```
mongoexport --db person --collection personData --  
type=csv --fields index,name --out per.csv
```

(Export to text file)

```
mongoexport --db person --collection personData --  
type=csv --fields index,name --out per.txt
```

- In Mongo database, the import utility imports the data from the Extended JSON or the CSV file created by the *mongoexport* command.
- The **mongoimport** command restores the documents from the JSON file into the Mongo collection.

A vertical collage of various digital and social media icons in shades of blue and white. The icons include a television, a lightbulb with rays, a camera, an ear, a smartphone, a shopping cart, a padlock, a briefcase, a heart, a bird (Twitter), and the letter 't' (Tumblr). The background is a dark blue grid pattern.

- Imports MongoDB collection data in JSON or CSV format
- Strict Mode or Shell Mode Extended JSON format is used

A vertical sidebar on the left side of the slide, featuring a grid of circular icons in shades of blue and white. The icons represent various concepts: a television, a camera, a lightbulb, an ear, a hand, a speech bubble, a padlock, a shopping cart, a smartphone, a person, a magnifying glass, a Twitter bird, and a lowercase 't' for Tumblr.

Example 1

```
mongoimport --db myDb --collection personsImport  
--file persons.json
```

```
mongoimport --host <hostname> --username
<username> --password <password> --
authenticationDatabase <authDb> --db
<database> --collection <newCollectionName>
--file <filename>
```

A decorative sidebar on the left side of the slide, featuring a grid of blue and white icons. The icons include a television, a camera, a lightbulb, a hand, a speech bubble, a padlock, a smartphone, a shopping cart, a Twitter bird, and a large letter 't'.

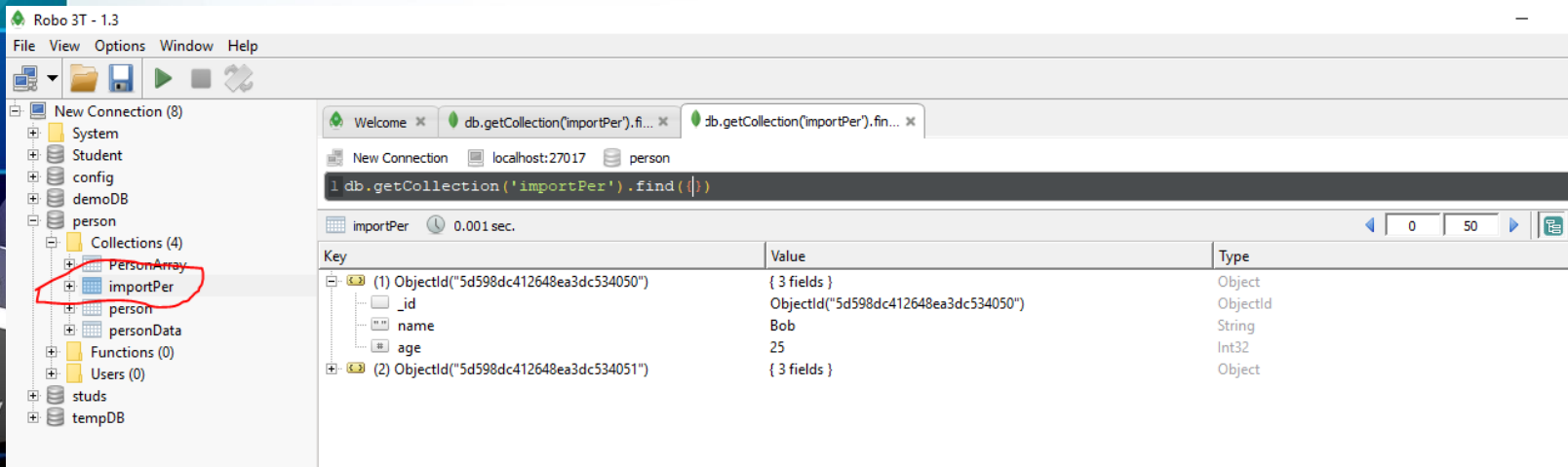
`mongodump`

- Binary export of the MongoDB database
- Without `--db` option all databases except 'local' will be dumped.
- Example:
 - `mongodump`

```
mongodump --host <hostname> --username  
<username> --password <password> --  
authenticationDatabase <authDb> --db  
<dumpDatabase> --collection <dumpCollection>  
--out <directoryName>
```

Example

- `mongoimport --db person --collection importPer --file newPersons.json`



Robo 3T - 1.3

File View Options Window Help

New Connection (8)

- System
- Student
- config
- demoDB
- person
 - Collections (4)
 - PersonArray
 - importPer**
 - person
 - personData
 - Functions (0)
 - Users (0)
- studs
- tempDB

Welcome x db.getCollection('importPer').find... x db.getCollection('importPer').find...

New Connection localhost:27017 person

```
1 db.getCollection('importPer').find({})
```

importPer 0.001 sec.

Key	Value	Type
(1) ObjectId("5d598dc412648ea3dc534050")	{ 3 fields }	Object
_id	ObjectId("5d598dc412648ea3dc534050")	ObjectId
name	Bob	String
age	25	Int32
(2) ObjectId("5d598dc412648ea3dc534051")	{ 3 fields }	Object



mongodb dump

- Binary export of the MongoDB database
- Without --db option all databases except 'local' will be dumped
- Examples

mongodump

mongodump --db myDb



Data backup -mongodb

- This command directly dumps all your data that are residing in your server into the dump directory.
- *mongodump* only detain the documents that are residing in your database as its backup data and does not contain index data.
- Then, *mongorestore* or *mongodh* have to rebuild the indexes once the data is restored.



mongodump Syntax

```
mongodump --host <hostname> --username  
<username> --password <password> --  
authenticationDatabase <authDb> --db  
<dumpDatabase> --collection <dumpCollection>  
--out <directoryName>
```

mongodump Compressed Archive

```
mongodump --host <hostname> --username  
<username> --password <password> --  
authenticationDatabase <authDb> --db  
<dumpDatabase> --collection <dumpCollection>  
--archive=<fileName> --gzip
```



- Initiate the MongoDB server using this command: *mongod*
- Then, in a new terminal/command prompt (cmd) in *administrative mode* initiate the MongoDB client system using the command: *mongo*
- Go to the necessary collection and execute the command: *mongodump*.
- As you run that command, you will see that all the data from the collection (e.g., my_project_db) got dumped.
- NOTE: Usually, mongodump comprehends the data from your database and then generates a BSON file where the data gets dumped.



mongorestore

- Binary import of the MongoDB backup
- Indexes will be re-created

- Examples

mongorestore



mongorestore Syntax

```
mongorestore --host <hostname> --username  
<username> --password <password> --  
authenticationDatabase <authDb> --dir  
<directoryName>
```



mongorestore Compressed Archive

```
mongorestore --host <hostname> --username  
<username> --password <password> --  
authenticationDatabase <authDb> --  
archive=<fileName> --gzip
```



mongostat

- MongoDB real-time statistics

- Examples

mongostat

mongostat

insert	query	update	delete	getmore	command	dirty	used	flushes	vsize	res	qrw	arw	net_in	net_out	conn	time
*0	*0	*0	*0	0	7 0	0.0%	1.0%	0	1.03G	102M	0 0	1 0	1.18k	66.1k	10 Apr 30	12:41:53.267
*0	*0	*0	*0	0	7 0	0.0%	1.0%	0	1.03G	102M	0 0	1 0	1.18k	66.0k	10 Apr 30	12:41:54.266
*0	*0	*0	*0	0	7 0	0.0%	1.0%	0	1.03G	102M	0 0	1 0	1.22k	65.4k	10 Apr 30	12:41:55.279
*0	*0	*0	*0	0	7 0	0.0%	1.0%	0	1.03G	102M	0 0	1 0	1.19k	66.8k	10 Apr 30	12:41:56.267
*0	*0	*0	*0	0	5 0	0.0%	1.0%	0	1.03G	102M	0 0	1 0	968b	64.0k	10 Apr 30	12:41:57.273
*0	*0	*0	*0	0	3 0	0.0%	1.0%	0	1.03G	102M	0 0	1 0	413b	59.5k	10 Apr 30	12:41:58.278
*0	*0	*0	*0	0	0 0	0.0%	1.0%	0	1.03G	102M	0 0	1 0	54b	20.2k	10 Apr 30	12:42:01.276



mongotop

- Top MongoDB current read and write operations

- Examples

mongotop

mongotop 180



Exporting Data from MongoDB

- ***mongoexport*** - through which users can export data from MongoDB databases to a JSON file format or a CSV file format.
- This utility is to be found in the MongoDB's bin subfolder (the path can be read like this: /mongodb/bin).
- As you run this feature, and while running, supply the database name, the name of the collection as well as the file you wish to export to, it performs the exporting task.



```
C:\Program Files\MongoDB\Server\4.0\bin> mongoexport --db person --collection  
personData --out persons  
2019-08-18T22:21:01.172+0530      connected to: localhost  
2019-08-18T22:21:01.268+0530      exported 16 records
```



Slide Title

Product A

- Feature 1
- Feature 2
- Feature 3

Product B

- Feature 1
- Feature 2
- Feature 3

