# MongoDBShell Documentation

***Release 1.0.11***

**Joe Drumgoole**

**May 02, 2019**

# Contents

Author : joe@joedrumgoole.com

Follow me on twitter like @jdrumgoole. for updates on this package.

MongoDBShell is a module that provides more natural interaction with MongoDB via the Python shell. Install using *pip3* (*MongoDBShell* only supports Python 3).

```
$pip3 install mongodbshell
```

To use:

```
>>> import mongodbshell
>>> client = mongodbshell.MongoDB()
>>> client.collection="test.test"
>>> client
mongodbshell.MongoDB('test', 'test', 'mongodb://localhost:27017')
>>> client.insert_one({"msg" : "MongoDBShell is great"})
ObjectId('5cb30cfa72a4ae3b105afa1c')
>>> client.find_one()
1    {'_id': ObjectId('5cb30cfa72a4ae3b105afa1c'), 'msg': 'MongoDBShell is great'}
>>> client.line_numbers = 0
>>> client.find_one()
{'_id': ObjectId('5cb30cfa72a4ae3b105afa1c'), 'msg': 'MongoDBShell is great'}
>>> # note the line number is no longer present
>>> client.output_file="output.txt" # send all output to this file
>>> client.find_one()
Output is also going to 'output.txt'
{'_id': ObjectId('5cb30cfa72a4ae3b105afa1c'), 'msg': 'MongoDBShell is great'}
>>> print(open("output.txt").read(), end="")
{'_id': ObjectId('5cb30cfa72a4ae3b105afa1c'), 'msg': 'MongoDBShell is great'}
>>>
```

This will give you a prebuilt `MongoDB` object.

**class** mongodbshell.**MongoDB**(*database_name='test'*, *collection_name='test'*, *host='mongodb://localhost:27017'*, **args*, ***kwargs*)

Simple command line MongoDB proxy for use in the Python shell.

**aggregate**(*pipeline*, *session=None*, ***kwargs*)

Run the aggregation pipeline

**client**

**Returns** the MongoDBClient object

**collection**

Assign to *collection* to reset the current default collection. Return the default collection object associated with the *MongoDB* object.

**collection_name**

**Returns** The name of the default collection

**collstats**(*scale=1024*, *verbose=False*)

Run collection stats for collection. see https://docs.mongodb.com/manual/reference/command/collStats/

**Parameters**

- **scale** – Scale at which to report sizes
- **verbose** – used for extended report on legacy MMAPV1 storage engine

**Returns** JSON doc with stats

**static confirm_yes**(*message*)
Return true if user confirms yes. A correct response is 'y' or 'Y'. All other chars will return false. :param message: A string :return: bool.

**count_documents**(*filter={}*, **args*, ***kwargs*)
Count all the documents in a collection accurately

**cursor_to_lines**(*cursor*, *format_func=None*)
Take a cursor that returns a list of docs and returns a generator yield each line of each doc a line at a time. :param cursor: A mongod cursor yielding docs (dictonaries) :param format_func: A customisable format function :return: a generator yielding a line at a time

**database**
Assign to this property to set the current default database. :return: Return the default database object associated with the Proxy

**database_name**

**Returns** The name of the default database

**dbstats**()
Run dbstats command for database See https://docs.mongodb.com/manual/reference/method/db.stats/

**delete_many**(**args*, ***kwargs*)
Run the pymongo delete_many command against the default database and collection and return the deleted IDs.

**delete_one**(**args*, ***kwargs*)
Run the pymongo delete_one command against the default database and collection and return the deleted IDs.

**doc_to_lines**(*doc*, *format_func=None*)
Generator that converts a doc to a sequence of lines. :param doc: A dictionary :param format_func: customisable formatter defaults to pformat :return: a generator yielding a line at a time

**find**(**args*, ***kwargs*)
Run the pymongo find command against the default database and collection and paginate the output to the screen.

**find_one**(**args*, ***kwargs*)
Run the pymongo find_one command against the default database and collection and paginate the output to the screen.

**insert_many**(**args*, ***kwargs*)
Run the pymongo insert_many command against the default database and collection and return the list of inserted IDs.

**insert_one**(**args*, ***kwargs*)
Run the pymongo insert_one command against the default database and collection and returne the inserted ID.

**is_master**()
Run the pymongo is_master command for the current server. :return: the is_master result doc.

**line_numbers**
Get and set the line_numbers boolean :return: *line_numbers* (TruelFalse)

**list_database_names**()
List all the databases on the default server.

**output_file**

**Returns** The name of the output file

**overlap**
Get and set the line_numbers boolean :return: *line_numbers* (TruelFalse)

**pager**(*lines*)
Outputs lines to a terminal. It uses *shutil.get_terminal_size* to determine the height of the terminal. It expects an iterator that returns a line at a time and those lines should be terminated by a valid newline sequence.

Behaviour is controlled by a number of external class properties.

*paginate* : Is on by default and triggers pagination. Without *paginate* all output is written straight to the screen.

*output_file* : By assigning a name to this property we can ensure that all output is sent to the corresponding file. Prompts are not output.

*pretty_print* : If this is set (default is on) then all output is pretty printed with *pprint*. If it is off then the output is just written to the screen.

*overlap* : The number of lines to overlap between one page and the next.

**Parameters** **lines** –

**Returns** paginated output

**paginate_doc**(*doc*)

**Parameters** **doc** – a dictionary of data

**Returns**

**pretty_print**
Get and set the pretty print boolean :return: *pretty_print* (TruelFalse)

**uri**

**Returns** The URI used to create the Proxy object

**static valid_mongodb_name**(*name*)
Check that the name for a database has no illegal characters :param name: the name of the database :return: True if the name is valid

**exception** mongodbshell.**MongoDBShellError**

**exception** mongodbshell.**ShellError**

CHAPTER 1

# Indices and tables

- genindex
- modindex
- search

# Python Module Index

## m

# Index