**Mongo Shell**

**Mongo shell** is an interactive JavaScript interface to MongoDB used to perform administrative operations and also query and update data.

Prior to connecting to MongoDB, ensure that MongoDB is running. If it is not, start MongoDB.

To start MongoDB, run the following command in a terminal.

**Windows**

```
C:\> "C:\Program Files\MongoDB\Server\4.0\bin\mongod.exe"
```

**Ubuntu**

```
$ sudo service mongod start
```

### Start Mongo Shell

Once you are sure that MongoDB is running,

**Windows**

Open another Command Window and run the following command.

```
C:\> "C:\Program Files\MongoDB\Server\4.0\bin\mongo.exe"
```

**Ubuntu**

Open another Terminal and run the following command to start mongo shell.

```
$ mongo
```
If the server is not running, you may get **connect failed** message as below:

```
$ mongo
MongoDB shell version v3.4.9
connecting to: mongodb://127.0.0.1:27017
```

When no parameters are provided with mongo command, the default functionality is that, the mongo shell tries to make a connection to the MongoDB server running at **localhost** on port **27017**.

But if you like to connect to MongoDB server that is running on a different machine connected to your network, you may make use of the options of mongo shell as shown below

```
$ mongo --host <host> --port <port_number>
```

An example is provided below

```
mongo --host 192.168.0.104 --port 28019
```

Now we shall run a simple query `db` to know the database the shell is pointing to.

```
> db
```
test is a default database.

Start Mongo Shell for a specified MongoDB instance among multiple instances

If you come across the scenario that there are multiple MongoDB instances running in a same machine, but on the different ports of-course, then to connect to a particular MongoDB instance differentiated by the port it is running on, run the following command:

```
$ mongo --port <port_number>
```

Following is an example demonstrating to open a Mongo Shell connected to a MongoDB instance running at 27018.

```
mongo --port 27018
```

Mongo Shell Commands

Following are a useful list of Mongo Shell Commands:

- `help` – show help
- `help admin` – administrative help
- `help connect` – connecting to a db help
- `help keys` – key shortcuts
- `help misc` – misc things to know
- `help mr` – mapreduce
- `show dbs` – show database names
- `show collections` – show collections in current database
- `show users` – show users in current database
- `show profile` – show most recent system.profile entries with time >= 1ms
- `show logs` – show the accessible logger names
- `show log [name]` – prints out the last segment of log in memory, ‘global’ is default
- `use <db_name>` – set current database
- `it` – result of the last line evaluated; use to further iterate
- `exit` – quit the mongo shell

Conclusion

In this MongoDB Tutorial, we have learnt about Mongo Shell, its usage and its connectivity to MongoDB Server when the server is running in another machine in network.
<table>
<thead>
<tr>
<th>Section</th>
<th>Subsections</th>
</tr>
</thead>
<tbody>
<tr>
<td>MongoDB Shell</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Check MongoDB Version</td>
</tr>
<tr>
<td></td>
<td>MongoDB Server Port Number - Default Value and How to Change it</td>
</tr>
<tr>
<td></td>
<td>Mongo Script</td>
</tr>
<tr>
<td>Database</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MongoDB Database</td>
</tr>
<tr>
<td></td>
<td>MongoDB Create Database</td>
</tr>
<tr>
<td></td>
<td>MongoDB Delete Database</td>
</tr>
<tr>
<td>Collections</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MongoDB Collection</td>
</tr>
<tr>
<td></td>
<td>MongoDB Create Collection</td>
</tr>
<tr>
<td></td>
<td>MongoDB Delete Collection</td>
</tr>
<tr>
<td>Documents</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MongoDB Document</td>
</tr>
<tr>
<td></td>
<td>MongoDB Insert Document</td>
</tr>
<tr>
<td></td>
<td>MongoDB Query Documents</td>
</tr>
<tr>
<td></td>
<td>MongoDB Project Fields in Result</td>
</tr>
<tr>
<td></td>
<td>MongoDB Update Document</td>
</tr>
<tr>
<td></td>
<td>MongoDB Delete Document</td>
</tr>
<tr>
<td></td>
<td>MongoDB Limit Documents</td>
</tr>
<tr>
<td></td>
<td>MongoDB Skip Documents</td>
</tr>
<tr>
<td></td>
<td>MongoDB Sort Documents</td>
</tr>
<tr>
<td></td>
<td>MongoDB Setup Replica Set</td>
</tr>
<tr>
<td></td>
<td>MongoDB Locks</td>
</tr>
<tr>
<td>MongoDB Concepts</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MongoDB Text Search</td>
</tr>
<tr>
<td></td>
<td>MongoDB MapReduce</td>
</tr>
<tr>
<td></td>
<td>MongoDB Backup - mongodump</td>
</tr>
<tr>
<td>MongoDB Queries</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MongoDB Date</td>
</tr>
<tr>
<td>MongoDB Integration</td>
<td></td>
</tr>
</tbody>
</table>
MongoDB Java
- Connect to MongoDB from Java

MongoDB Python
- Connect to MongoDB from Python

MongoDB Kotlin
- Connect to MongoDB from Kotlin

MongoDB Node.js
- Node.js MongoDB
  - Node.js MongoDB Connection
  - Node.js MongoDB Create Database
  - Node.js MongoDB Drop Database
  - Node.js MongoDB Create Collection
  - Node.js MongoDB Delete Collection
  - Node.js MongoDB Insert Documents
  - MongoError: failed to connect to server

MongoDB Others
- MongoDB Interview Questions
- Uninstall MongoDB from Ubuntu

Useful Resources
- How to Learn Programming