The mongo Shell

**On this page**

* [Introduction](https://docs.mongodb.org/manual/mongo/#introduction)
* [Start the mongo Shell](https://docs.mongodb.org/manual/mongo/#start-the-mongo-shell)
* [Working with the mongo Shell](https://docs.mongodb.org/manual/mongo/#working-with-the-mongo-shell)
* [Tab Completion and Other Keyboard Shortcuts](https://docs.mongodb.org/manual/mongo/#tab-completion-and-other-keyboard-shortcuts)
* [Exit the Shell](https://docs.mongodb.org/manual/mongo/#exit-the-shell)

Introduction

The [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell is an interactive JavaScript interface to MongoDB. You can use the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell to query and update data as well as perform administrative operations.

The [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell is a component of the [MongoDB distributions](http://www.mongodb.org/downloads). Once you have [installed and have started MongoDB](https://docs.mongodb.org/manual/installation/), connect the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell to your running MongoDB instance.

Most examples in the [MongoDB Manual](https://docs.mongodb.org/manual/) use the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell; however, many [drivers](https://docs.mongodb.org/manual/applications/drivers/) provide similar interfaces to MongoDB.

Start the mongo Shell

**IMPORTANT**

Ensure that MongoDB is running before attempting to start the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell.

To start the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell and connect to your [MongoDB](https://docs.mongodb.org/manual/reference/program/mongod/) instance running on **localhost** with **default port**:

1. At a prompt in a terminal window (or a command prompt for Windows), go to your <mongodbinstallation dir>:
2. cd <mongodb installation dir>
3. Type ./bin/mongo to start [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo):
4. ./bin/mongo

If you have added the <mongodb installation dir>/bin to the PATH environment variable, you can just type mongo instead of ./bin/mongo.

**Options**

When you run [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) without any arguments, the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell will attempt to connect to the MongoDB instance running on the localhost interface on port 27017. To specify a different host or port number, as well as other options, see [examples of starting up mongo](https://docs.mongodb.org/manual/reference/program/mongo/#mongo-usage-examples) and [mongo reference](https://docs.mongodb.org/manual/reference/program/mongo/) which provides details on the available options.

**.mongorc.js File**

When starting, [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) checks the user’s [HOME](https://docs.mongodb.org/manual/reference/program/mongo/#envvar-HOME) directory for a JavaScript file named [.mongorc.js](https://docs.mongodb.org/manual/reference/program/mongo/#mongo-mongorc-file). If found,[mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) interprets the content of .mongorc.js before displaying the prompt for the first time. If you use the shell to evaluate a JavaScript file or expression, either by using the *--eval* option on the command line or by specifying [a .js file to mongo](https://docs.mongodb.org/manual/reference/program/mongo/#mongo-shell-file), [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) will read the .mongorc.js file *after* the JavaScript has finished processing. You can prevent .mongorc.js from being loaded by using the [--norc](https://docs.mongodb.org/manual/reference/program/mongo/#cmdoption--norc) option.

Working with the mongo Shell

To display the database you are using, type db:

db

The operation should return test, which is the default database. To switch databases, issue the use <db>helper, as in the following example:

use <database>

To list the available databases, use the helper show dbs. See also [db.getSiblingDB()](https://docs.mongodb.org/manual/reference/method/db.getSiblingDB/" \l "db.getSiblingDB" \o "db.getSiblingDB()) method to access a different database from the current database without switching your current database context (i.e.db).

You can switch to non-existing databases. When you first store data in the database, such as by creating a collection, MongoDB creates the database. For example, the following creates both the databasemyNewDatabase and the [collection](https://docs.mongodb.org/manual/reference/glossary/#term-collection) myCollection during the [insert()](https://docs.mongodb.org/manual/reference/method/db.collection.insert/#db.collection.insert) operation:

use myNewDatabase

db.myCollection.insert( { x: 1 } );

The [db.myCollection.insert()](https://docs.mongodb.org/manual/reference/method/db.collection.insert/" \l "db.collection.insert" \o "db.collection.insert()) is one of the [methods available in the mongo shell](https://docs.mongodb.org/manual/reference/method/)

* db refers to the current database.
* myCollection is the name of the collection.

If the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell does not accept the name of the collection, for instance if the name contains a space, hyphen, or starts with a number, you can use an alternate syntax to refer to the collection, as in the following:

db["3test"].find()

db.getCollection("3test").find()

For more documentation of basic MongoDB operations in the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell, see:

* [Getting Started Guide](https://docs.mongodb.org/getting-started/shell)
* [Insert Documents](https://docs.mongodb.org/manual/tutorial/insert-documents/)
* [Query Documents](https://docs.mongodb.org/manual/tutorial/query-documents/)
* [Modify Documents](https://docs.mongodb.org/manual/tutorial/modify-documents/)
* [Remove Documents](https://docs.mongodb.org/manual/tutorial/remove-documents/)
* [mongo Shell Methods](https://docs.mongodb.org/manual/reference/method/)

**Format Printed Results**

The [db.collection.find()](https://docs.mongodb.org/manual/reference/method/db.collection.find/" \l "db.collection.find" \o "db.collection.find()) method returns a [cursor](https://docs.mongodb.org/manual/reference/glossary/#term-cursor) to the results; however, in the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell, if the returned cursor is not assigned to a variable using the var keyword, then the cursor is automatically iterated up to 20 times to print up to the first 20 documents that match the query. The [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell will prompt Typeit to iterate another 20 times.

To format the printed result, you can add the .pretty() to the operation, as in the following:

db.myCollection.find().pretty()

In addition, you can use the following explicit print methods in the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell:

* print() to print without formatting
* print(tojson(<obj>)) to print with [JSON](https://docs.mongodb.org/manual/reference/glossary/#term-json) formatting and equivalent to printjson()
* printjson() to print with [JSON](https://docs.mongodb.org/manual/reference/glossary/#term-json) formatting and equivalent to print(tojson(<obj>))

For more information and examples on cursor handling in the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell, see [Cursors](https://docs.mongodb.org/manual/core/cursors/). See also [Cursor Help](https://docs.mongodb.org/manual/tutorial/access-mongo-shell-help/#mongo-shell-help-cursor) for list of cursor help in the [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell.

**Multi-line Operations in the mongo Shell**

If you end a line with an open parenthesis ('('), an open brace ('{'), or an open bracket ('['), then the subsequent lines start with ellipsis ("...") until you enter the corresponding closing parenthesis (')'), the closing brace ('}') or the closing bracket (']'). The [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell waits for the closing parenthesis, closing brace, or the closing bracket before evaluating the code, as in the following example:

> **if** ( x > 0 ) {

... count++;

... print (x);

... }

You can exit the line continuation mode if you enter two blank lines, as in the following example:

> **if** (x > 0

...

...

>

Tab Completion and Other Keyboard Shortcuts

The [mongo](https://docs.mongodb.org/manual/reference/program/mongo/#bin.mongo) shell supports keyboard shortcuts. For example,

* Use the up/down arrow keys to scroll through command history. See [.dbshell](https://docs.mongodb.org/manual/reference/program/mongo/#mongo-dbshell-file) documentation for more information on the .dbshell file.
* Use <Tab> to autocomplete or to list the completion possibilities, as in the following example which uses <Tab> to complete the method name starting with the letter 'c':
* db.myCollection.c<Tab>

Because there are many collection methods starting with the letter 'c', the <Tab> will list the various methods that start with 'c'.

For a full list of the shortcuts, see [Shell Keyboard Shortcuts](https://docs.mongodb.org/manual/reference/program/mongo/#mongo-keyboard-shortcuts)

Exit the Shell

To exit the shell, type quit() or use the <Ctrl-c> shortcut.