mongoDB

- *open-source*
- *high-performance*
- *document-oriented database*

What are Commands?

Commands are special MongoDB operations that are executed by running a query against the ***$cmd*** collection. Most drivers provide a helper for running commands; here's an example of how to run the dropDatabase command from the shell:

```
> db.runCommand({dropDatabase:1});
```

Some commands are admin-only, and must be run on the ***admin*** database.
In the list below those commands are marked with an asterisk (*).

Available Commands

To get a list of all commands available on your version of the server, run **"mongod --rest"** and then visit **http://localhost:28017/_commands** in your web browser.

Some of the most frequently used commands are listed below:

`*{buildinfo: 1}`	**Get version number and other build information about the MongoDB server.**
`{collStats: coll[,scale: 1]}`	**Get stats about collection *coll*. Sizes are in bytes / *scale*.**
`{count: coll[,query: query]}`	**Get the number of documents in collection *coll* that match the (optional) specified *query*.**
`{dbStats: 1}`	**Get stats about the current database.**
`{distinct: coll, key: key[,query: query]}`	**Get a list of distinct values for *key* in *coll* for all documents that match the (optional) specified *query*.**
`{drop: coll}`	**Delete collection *coll*.**

`{dropDatabase: 1}`	**Delete the current database.**
`{dropIndexes: coll, name: name}`	**Drop the index named *name* in collection *coll*. Use the name "*" to drop all indexes in *coll*.**
`{findAndModify: ...}`	**Perform a findAndModify operation.**
`*{fsync: ...}`	**Perform a full fsync and optionally take a write lock.**
`{getLastError: 1}`	**Get the status of the last operation on this connection.**
`{group: ...}`	**Perform a grouping aggregation.**
`{isMaster: 1}`	**Check if this server is a primary/master server.**
`{listCommands: 1}`	**Get a list of available commands.**
`*{listDatabases: 1}`	**Get a list of databases on this server.**
`{mapReduce: ...}`	**Run a map/reduce job**
`{profile: n}`	**Set the database profiler to profiling level *n*.**
`{reIndex: coll}`	**Re-index collection *coll*.**
`*{renameCollection: a, to: b}`	**Rename collection *a* to *b*.**
`{repairDatabase: 1}`	**Repair and compact the current database - can be slow.**
`*{replSetGetStatus: 1}`	**Get the status of a replica set.**
`*{replSetInitiate: config}`	**Initiate a new replica set with *config*.**
`*{replSetReconfig: config}`	**Reconfigure a replica set with *config*.**
`*{replSetStepDown: 1}`	**If primary in a replica set, step down and let a secondary be promoted.**
`*{resync: 1}`	**Start a full resync on a replica slave.**
`{serverStatus: 1}`	**Get lots of administrative statistics about the server.**
`*{shutdown: 1}`	**Shutdown the MongoDB server.**
`*{top: 1}`	**Get a breakdown of usage by collection.**
`{validate: ns}`	**Validate the namespace (collection or index) *ns* - can be slow.**