# MongoDB CLI Command Reference

## 1. Connecting and Basic CLI Commands

* mongosh

→ Start the MongoDB shell.

* mongosh "mongodb://localhost:27017"

→ Connect to a MongoDB instance.

* show dbs

→ List all databases.

* use <database\_name>

→ Switch to (or create) a database.

* db

→ Display the current database.

* show collections

→ List all collections in the current database.

* exit

→ Exit the Mongo shell.

## 2. Database Operations

* db.createCollection('students')

→ Create a new collection.

* db.dropDatabase()

→ Delete the current database.

* db.getName()

→ Get the name of the current database.

* db.stats()

→ Show statistics for the current database.

* db.getCollectionNames()

→ List all collections in the database.

## 3. Collection Operations

* db.createCollection('products')

→ Create a collection.

* db.products.drop()

→ Delete a collection.

* db.products.stats()

→ Get statistics about a collection.

* db.products.renameCollection('items')

→ Rename a collection.

## 4. Insert Documents

* db.users.insertOne({ name: 'Alice', age: 25 })

→ Insert one document.

* db.users.insertMany([{ name: 'Bob', age: 30 }, { name: 'Eve', age: 28 }])

→ Insert multiple documents.

* db.users.find()

→ Verify inserted documents.

## 5. Query Documents

* db.users.find()

→ Show all documents.

* db.users.find().pretty()

→ Pretty-print results.

* db.users.findOne({ name: 'Alice' })

→ Find one document.

* db.users.find({ age: { $gt: 25 } })

→ Find documents where age > 25.

* db.users.find({ age: { $gte: 25, $lte: 30 } })

→ Between 25 and 30.

* db.users.find({ name: { $in: ['Alice', 'Bob'] } })

→ Match names in a list.

* db.users.find({}, { name: 1, \_id: 0 })

→ Show only name field.

* db.users.countDocuments()

→ Count total documents.

## 6. Update Documents

* db.users.updateOne({ name: 'Alice' }, { $set: { age: 26 } })

→ Update one record.

* db.users.updateMany({ department: 'IT' }, { $set: { status: 'Active' } })

→ Update many records.

* db.users.replaceOne({ name: 'Bob' }, { name: 'Bobby', age: 31 })

→ Replace a document.

* db.users.find()

→ Verify updates.

## 7. Delete Documents

* db.users.deleteOne({ name: 'Eve' })

→ Delete one record.

* db.users.deleteMany({ department: 'Sales' })

→ Delete multiple records.

* db.users.remove({})

→ Remove all documents (deprecated, use deleteMany).

## 8. Indexing

* db.users.createIndex({ name: 1 })

→ Create ascending index on name.

* db.users.createIndex({ age: -1 })

→ Create descending index on age.

* db.users.getIndexes()

→ List all indexes.

* db.users.dropIndex({ name: 1 })

→ Drop a specific index.

## 9. User & Role Management (Admin)

* use admin

→ Switch to admin DB.

* db.createUser({ user: 'admin', pwd: 'pass123', roles: ['root'] })

→ Create admin user.

* db.createUser({ user: 'appUser', pwd: 'pass123', roles: [{ role: 'readWrite', db: 'mydb' }] })

→ Create a read/write user.

* db.dropUser('appUser')

→ Delete a user.

* db.getUsers()

→ List all users.

## 10. Aggregation and Grouping

* db.sales.aggregate([{ $group: { \_id: '$region', totalSales: { $sum: '$amount' } } }])

→ Group by region and sum sales.

* db.sales.aggregate([{ $match: { region: 'East' } }])

→ Filter before grouping.

* db.sales.aggregate([{ $sort: { totalSales: -1 } }])

→ Sort results descending.

* db.sales.aggregate([{ $project: { \_id: 0, region: 1, totalSales: 1 } }])

→ Include specific fields.

## 11. Backup & Restore (Shell Tools)

* mongodump --db mydb --out backup/

→ Backup database.

* mongorestore --db mydb backup/mydb/

→ Restore database.

* mongoexport --db mydb --collection users --out users.json

→ Export collection to JSON.

* mongoimport --db mydb --collection users --file users.json

→ Import collection from JSON.

## 12. System and Performance

* db.serverStatus()

→ Show server performance stats.

* db.stats()

→ Database stats.

* db.collection.stats()

→ Collection stats.

* db.currentOp()

→ Show running operations.

* db.killOp(<opid>)

→ Kill a running operation.

## Bonus: Useful Shortcuts

* cls

→ Clear the Mongo shell screen (Windows).

* Ctrl + L

→ Clear screen (Linux/macOS).

* help

→ Show available Mongo shell help.

* db.help()

→ Show help for database-level commands.

* db.collection.help()

→ Show help for collection-level commands.