**MongoDB**

[Getting Started 1](#_Toc178863349)

[Terms to remember 1](#_Toc178863350)

# Getting Started

1. Install MongoDB Community server
2. Install MongoDB Shell
3. Install MongoDB Compass (GUI)

# Terms to remember

1. Cluster - Database
2. Collection - Table
3. Document – record

# Commands

## Terminal commands

mongosh – To open the mongodb shell in another terminal.

cls – clear screen

// Note: We do not need to use a ";" to terminate a command.

show dbs - // shows all the databases in the server.

use <database\_name> - // switch to the database specified. Works even if the database specified does not exist.

db - // to show the current database

show collections - // Show the tables in the database.

var name = "Daniel" - // To create a variable and initialise it.

name = // to return the value of the variable created

help - // List all the commands you can use and what they mean

exit - // To exit the shell

## CRUD Commands

### Create

db.books.insertOne({title: "The River and the source", author: "Margaret Ogola", pages: 281, genres: [   "Fiction", "Historical"], rating: 4.8})

-- To insert one document/record into the collection/table

db.books.insertMany([{title: "Kidagaa Kimemwozea", author: "Ken Walibora", pages: 281, genres: ["Fiction", "Romance", "Politics"], rating: 4.8}, {title: "Caucasian Chalk Circle", author: "Bertolt Bretch", pages: 281, genres: ["Fiction", "Romance", "Politics"], rating: 4.8}])

-- To insert many documents/records into the collection/table

### Read

db.books.find() -- Returns the first 20 documents in the collection

it – To iterate over the next 20 documents

db.books.find({author: "Margaret Ogola"}) – To filter the documents by one property

db.books.find({ author: "Margaret Ogola", pages: 281 })"}) – To filter the documents by more than one property

db.books.find({author: "Margaret Ogola"}, {title: 1, pages: 1, rating: 1}) -- Returns the specified "title", "pages" and "rating" properties of the documents in the collection, that match the the filter passeed

db.books.findOne({title: 'The River and the source'}) -- Returns the first document that passes the filter specified