University of El Oued

Faculty of Exact Sciences

Department of Computer Science

Lab Report 5: NoSQL Databases

MongoDB Setup & CRUD Operations

Student:
Mohammed Seddik Lifa

Specialization:
Master II: AI & Data Science

Date:
11-10-2025

1 Objective

The primary objective of this lab is to install MongoDB, configure the background service, and perform fundamental Create, Read, Update, and Delete (CRUD) operations using the modern `mongosh` client. This report demonstrates the successful execution of these tasks.

2 Implementation & Proof of Execution

2.1 A. Create a Database

Instruction: Create and switch to a new database named `info` using the command `use info`.

Execution Proof: The shell successfully switched context to the new database.

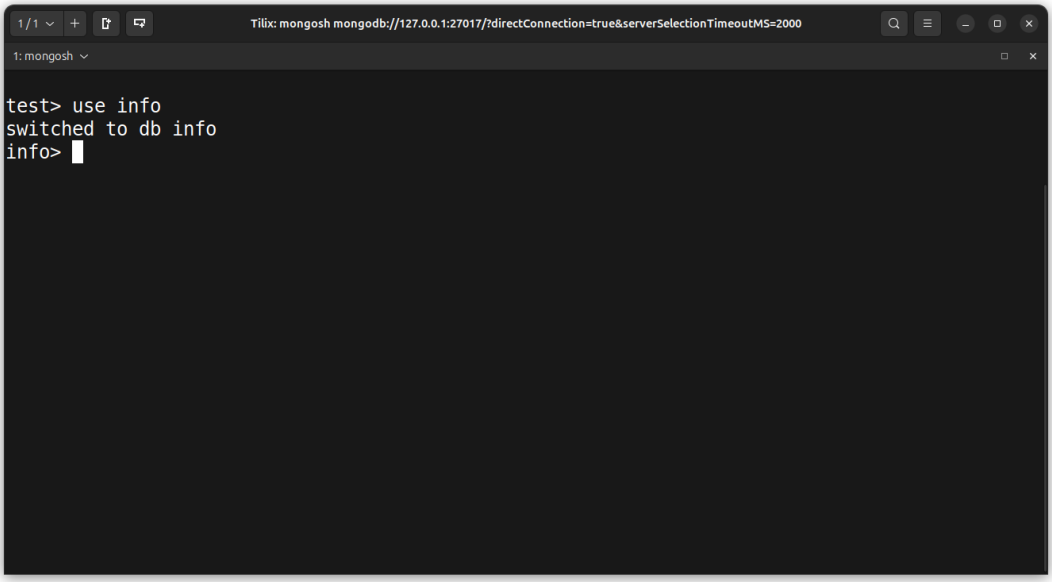

Figure 1: Command execution: Switching to database 'info'

2.2 B. Create a Collection (Insert Documents)

Instruction: Insert a dataset into the `produits` collection. The dataset includes details (manufacturer, price, and options) for a Macbook Pro, Macbook Air, and Thinkpad X230.

Code Pattern Executed:

```
db.produits.insertMany([
  { nom: "Macbook Pro", fabriquant: "Apple", prix: 11435.99 ... },
  { nom: "Macbook Air", fabriquant: "Apple", ultrabook: true ... },
  { nom: "Thinkpad X230", fabriquant: "Lenovo", ultrabook: true ... }
```

```
])
```

Listing 1: Insertion Query Structure

Execution Proof: The output shows `acknowledged: true` and three generated ObjectIDs.

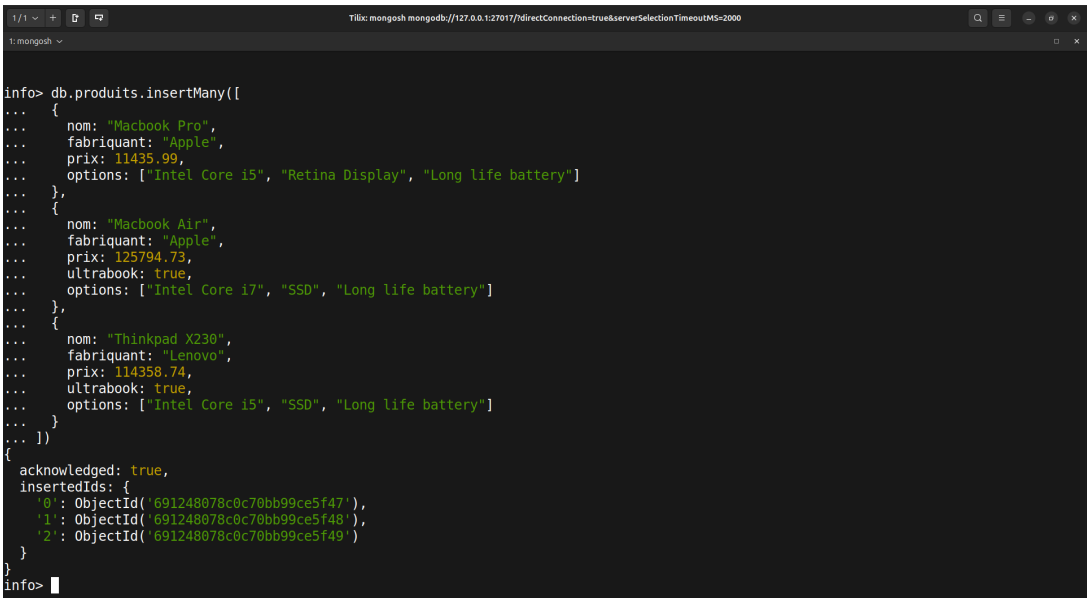

Figure 2: Successful insertion of 3 documents with ObjectIDs

2.3 C. Read (Query) Documents

Instruction: Perform various queries to retrieve data. Specifically:

1. Retrieve a single document.
2. Filter documents where `ultrabook` is true.
3. Use Regex to find products starting with "Macbook".

Queries Run:

```
db.produits.findOne()
db.produits.findOne({ ultrabook: true })
db.produits.find({ nom: { $regex: "Macbook" } })
```

Execution Proof:

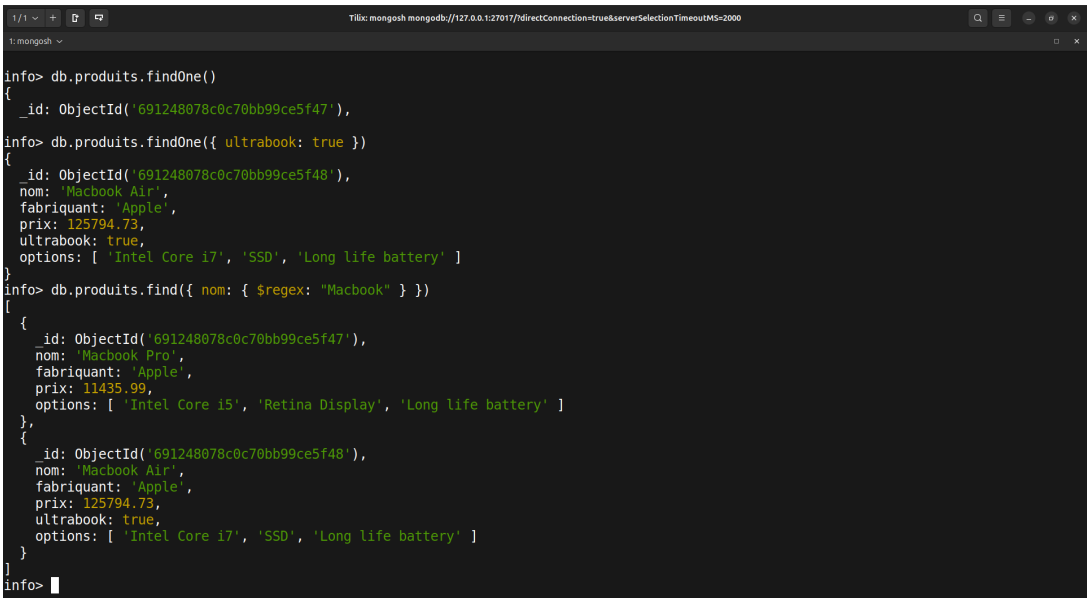

Figure 3: Query Results: findOne, Boolean filtering, and Regex matching

2.4 D. Delete Documents

Instruction: Remove specific data from the collection:

- Delete all documents where manufacturer is "Apple".
- Delete a specific document by its unique `_id`.

Queries Run:

```
// Delete all Apple products
db.produits.deleteMany({ fabriquant: "Apple" })

```

```
// Delete specific ID (Thinkpad)
db.produits.deleteOne({ _id: thinkpad_id })

// Verify Empty Collection
db.produits.find()
```

Execution Proof: The output confirms `deletedCount: 2` for Apple products and `deletedCount: 1` for the Thinkpad. The final find command returns nothing, confirming the collection is empty.

Figure 4: Deletion operations and final verification