**[CS-340-12415-M01 Client/Server Development 2025 C](https://learn.snhu.edu/d2l/home/1831816" \o "CS-340-12415-M01 Client/Server Development 2025 C-1 (Jan - Mar))**

**Module 2: 2-1 Assignment: CRUD in MongoDB**

**Create, Read, Update, and Delete(CRUD)**

**CS 340 Module Two Assignment**

**Marissa Lanza**

**1/17/2025**

**Create, Read, Update, and Delete To-Do List:**

1. **Load Dataset**:
   * Access the directory and load the dataset into MongoDB using the mongoimport tool:

bash

Copy

cd /usr/local/datasets/

mongoimport --username="${MONGO\_USER}" --password="${MONGO\_PASS}" --port=${MONGO\_PORT} --host=${MONGO\_HOST} --db city --collection inspections --authenticationDatabase admin --drop ./city\_inspections.json

1. **Verify Data Load**:
   * Use MongoDB shell commands to verify the data:

javascript

Copy

db.inspections.find({"id" : "10021-2015-ENFO"})

db.inspections.find({"result":"Out of Business"},{"business\_name":1}).limit(10)

* + Take screenshots of the results.

1. **Insert New Document**:
   * Insert the provided document into the inspections collection:

javascript

Copy

db.inspections.insertOne({

"id": "20032-2020-ACME",

"certificate\_number": 9998888,

"business\_name": "ACME Explosives",

"date": new Date(),

"result": "Business Padlocked",

"sector": "Explosive Retail Dealer-999",

"address": { "number": 1721, "street": "Boom Road", "city": "BRONX", "zip": "10463" }

})

* + Verify the insertion with findOne() and provide a screenshot.

1. **Answer Questions**:
   * Use MongoDB queries for the following: a. List distinct "sector" and count. b. Compare the date data type for "AUSTIN 2012" and "ACME Explosives". c. Count businesses with a "Violation Issued".
2. **Update Document**:
   * Update the business document with the ID 20032-2020-ACME:

javascript

Copy

db.inspections.updateOne(

{ "id": "20032-2020-ACME" },

{ $set: { "business\_name": "New ACME Flowers", "result": "Business Re-opened", "comments": "Flowers after the explosion" } }

)

* + Provide a screenshot of the update.

1. **Perform Updates**:
   * Update all documents with city: "ROSEDALE" in the address to set the zip code to "76114".

javascript

Copy

db.inspections.updateMany(

{ "address.city": "ROSEDALE" },

{ $set: { "address.zip": "76114" } }

)

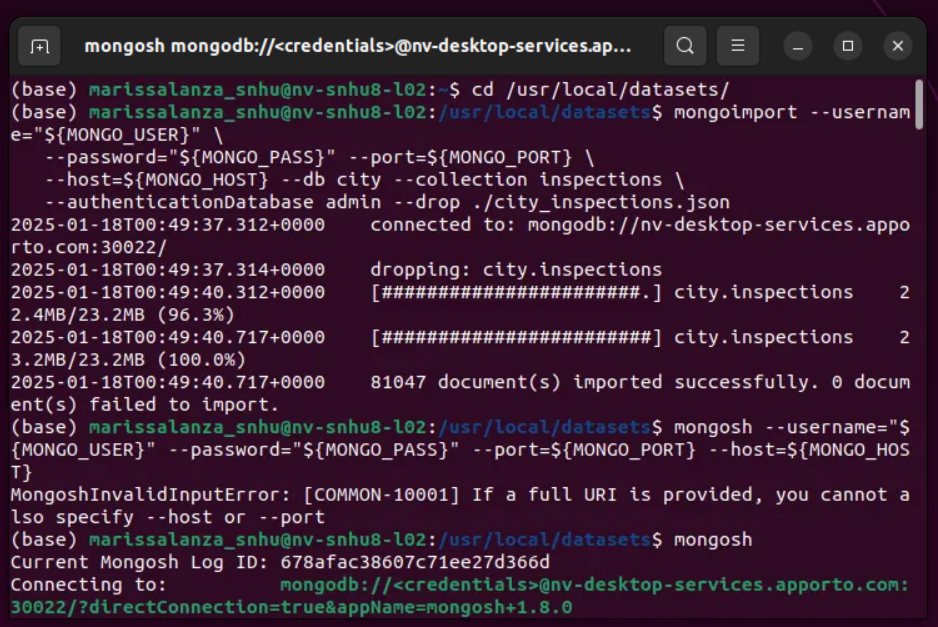
* + Remove the first document with "result": "Violation Issued":

javascript

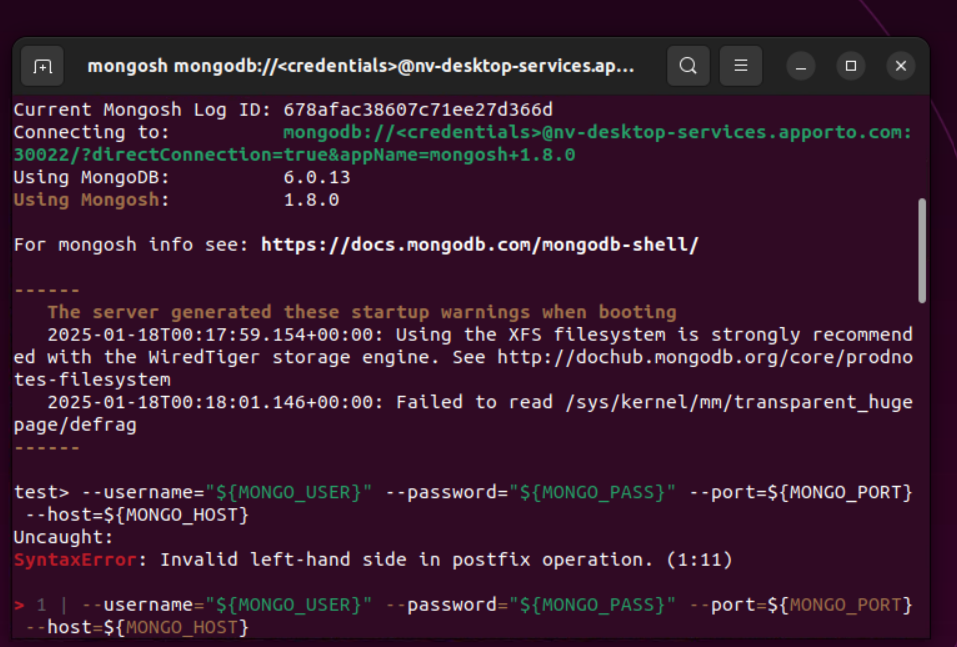
Copy

db.inspections.deleteOne({ "result": "Violation Issued" })

1. **Provide Screenshots**:
   * Take screenshots of the results of your queries and updates for submission.

**Screenshot 1:**

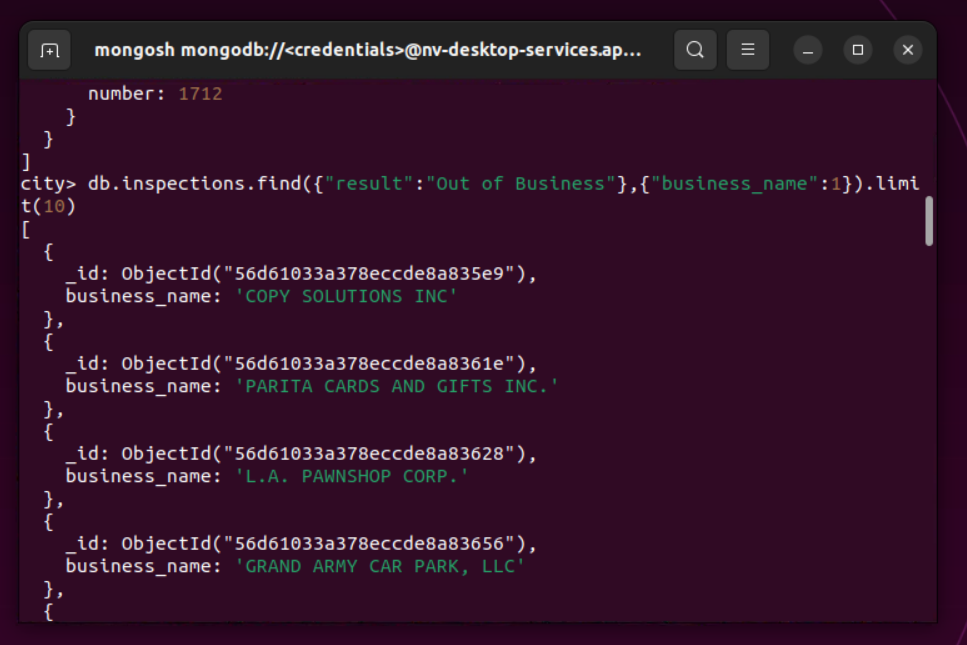
***This screenshot shows the command cd /usr/local/datasets/ used to change into the Apporto directory containing the datasets, followed by the use of the mongoimport utility to import the city\_inspections.json file into MongoDB. It also shows the successful import of the dataset into the inspections collection of the city database.***

***Screenshot 2: The error displayed in the screenshot is a SyntaxError related to the mongo shell command. Specifically, the message Invalid left-hand side in postfix operation suggests that there is an issue with the way the variables are being used in the command.***

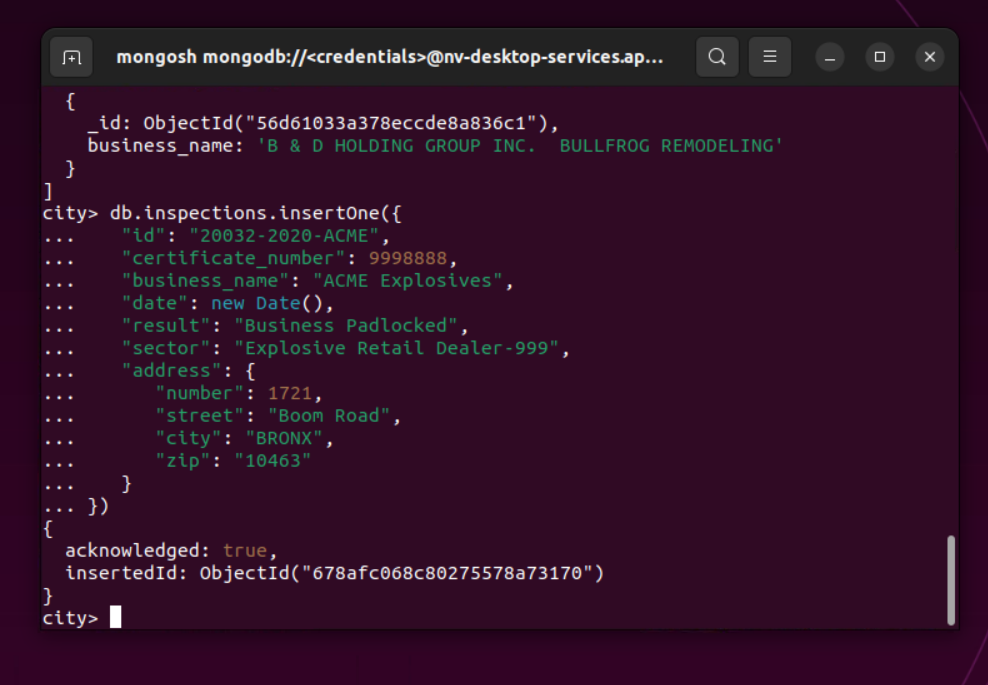
A screenshot of a computer program

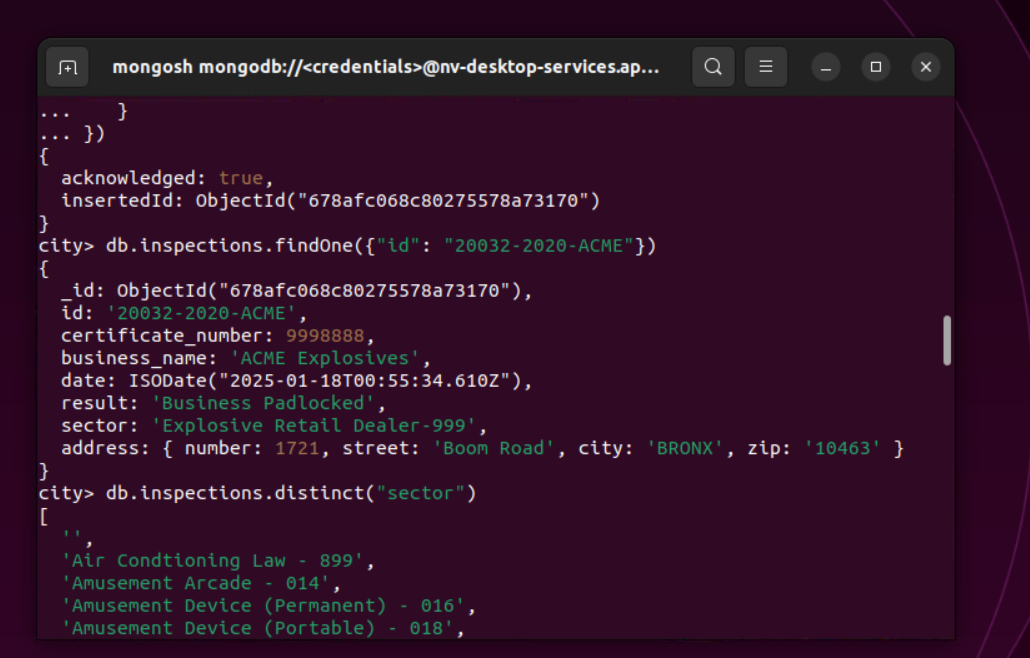
Description automatically generated

***Screenshot 3: This screenshot shows the successful query of the "inspections" collection in the "city" database, retrieving a document with the ID "10021-2015-ENFO" and displaying its details, such as business name, result, and address. The query verifies that the data was loaded correctly.***



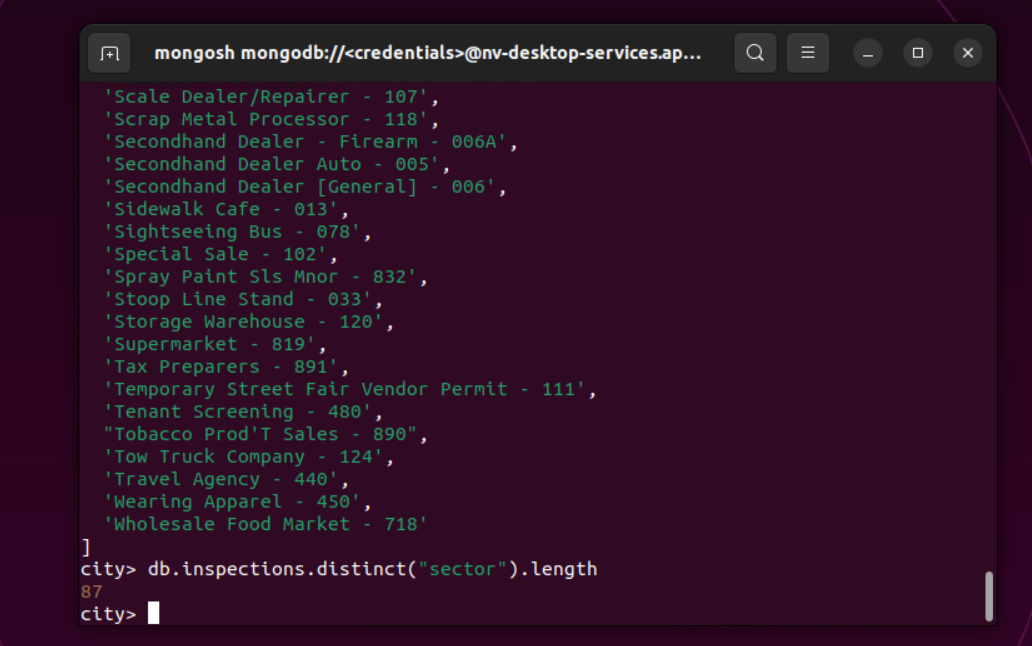
***Screenshot 4: This screenshot displays the query result from the "inspections" collection of the "city" database. It lists businesses marked as "Out of Business" with their names, such as "COPY SOLUTIONS INC," "PARITA CARDS AND GIFTS INC.," "L.A. PAWNSHOP CORP.," and "GRAND ARMY CAR PARK, LLC." The query only returns the business names from the first 10 results.***

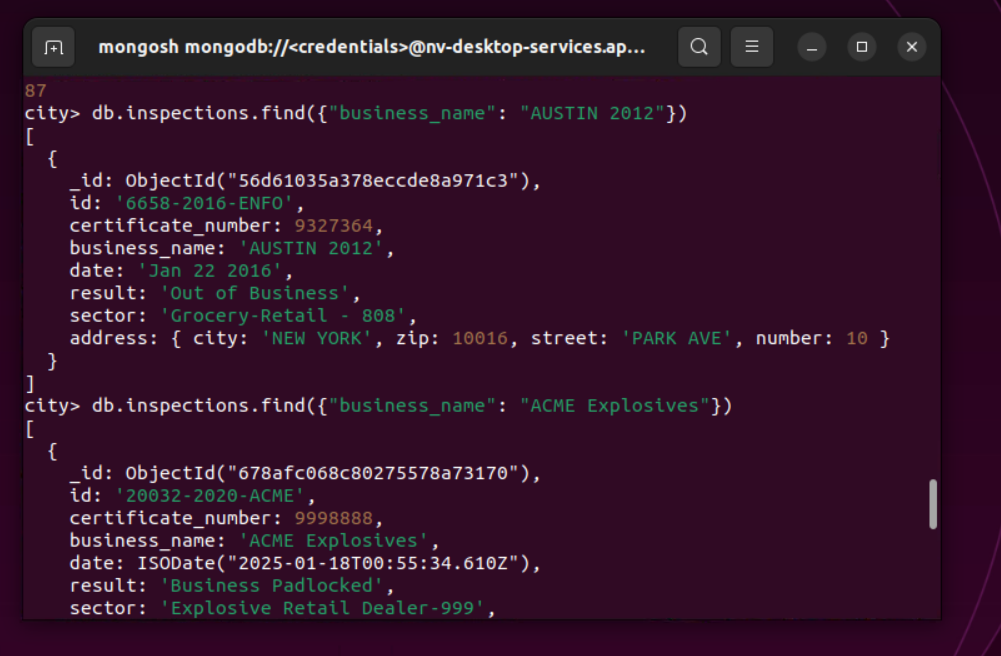
******

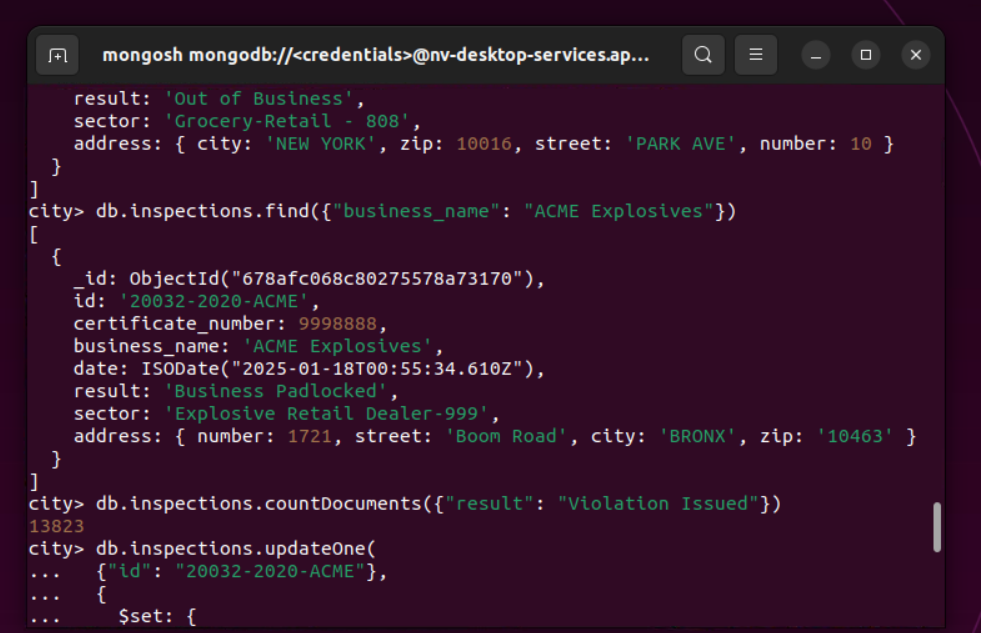
***This screenshot 5 above: shows the successful insertion of a new document into the "inspections" collection in the "city" database. The document includes details for a business named "ACME Explosives," such as its ID, certificate number, business name, inspection result, sector, and address. The operation was successful, with a new ObjectId generated for the document.***

***This screenshot6: shows the result of two MongoDB commands:***

1. ***db. inspections.findOne({"id": "20032-2020-ACME"}) - This command retrieves the document for "ACME Explosives" with details such as certificate number, business name, result, sector, and address.***
2. ***db. inspections. distinct("sector") - This command returns the distinct list of sectors in the "inspections" collection. The screenshot includes sectors like "Air Conditioning Law" and "Amusement Arcade" among others.***

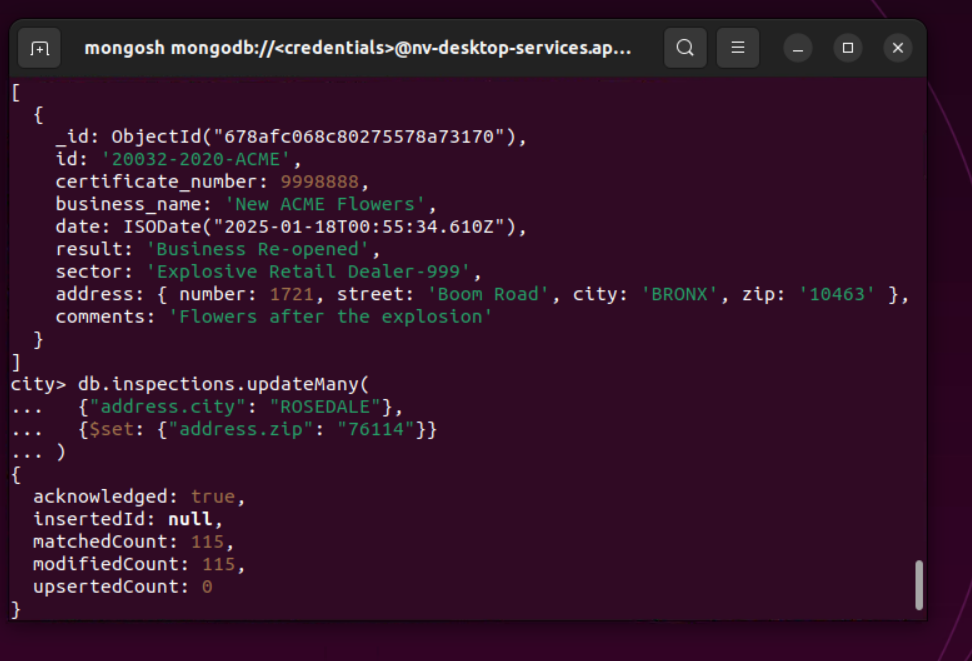
***This screenshot7: shows the distinct sectors query in MongoDB, returning a list of 87 unique sectors in the "inspections" collection of the "city" database. The command db. inspections. distinct("sector"). length was used to retrieve and count the sectors.***

***This screenshot 8: shows MongoDB queries for retrieving details about businesses with the names "AUSTIN 2012" and "ACME Explosives." The queries return documents with information such as the business name, certificate number, inspection result, and address.***

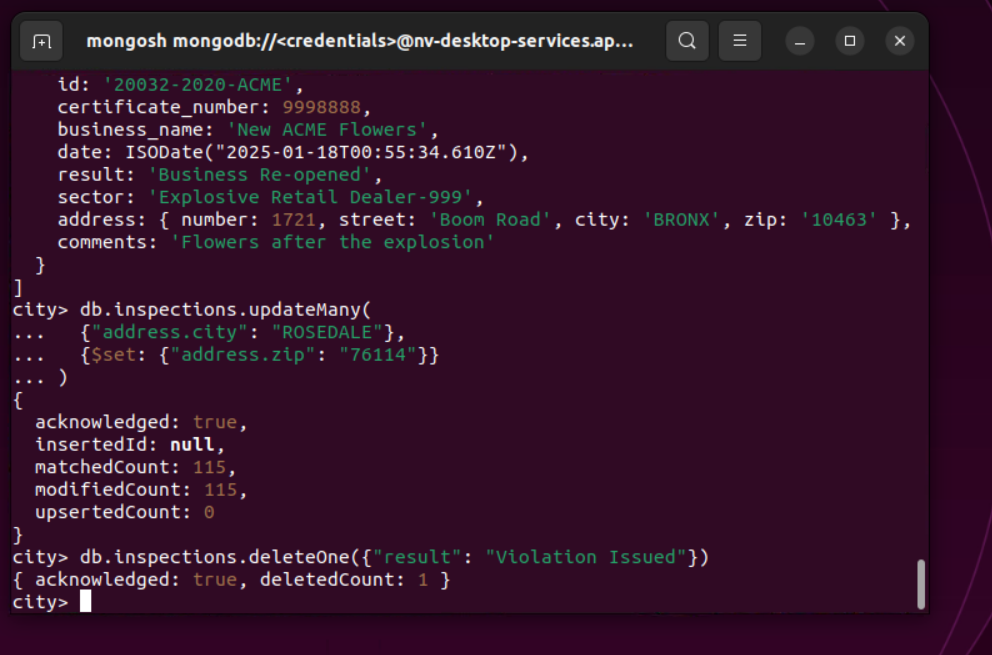


***This screenshot 9: shows MongoDB queries for the "ACME Explosives" business, including retrieving its information by business name, counting documents with the "Violation Issued" result, and updating the document with the business ID "20032-2020-ACME." It also demonstrates an update to the document with the new details for the business, such as the "result" field being changed to "Violation Issued”.***

***This screenshot shows the update of the "ACME Explosives" document in MongoDB, changing the business name to "New ACME Flowers," the result to "Business Re-opened," and adding a comment "Flowers after the explosion." The screenshot also shows the query to verify the update by finding the document with the ID "20032-2020-ACME." The update is acknowledged, with the matched and modified counts indicating the successful update.***

******

***This screenshot 11: shows the successful update of multiple documents in MongoDB using the updateMany command. The command updates the "zip" field for all documents with the "city" field set to "ROSEDALE" to a new zip code "76114." The update is acknowledged, with 115 documents matched and modified as shown in the acknowledgment message.***

******

***This screenshot 12: shows two MongoDB operations. The first operation uses updateMany to update the zip code for all documents where the "city" is "ROSEDALE," setting the zip code to "76114." The second operation uses deleteOne to delete the first document where the "result" field is "Violation Issued," with one document being successfully deleted.***