MongoDB Weekly Task

**1. Create Operation**

Query:

use library

**Explanation:** This command creates or switches to the database named `library`.

Query:

db.createCollection("books")

**Explanation:** This creates the `books` collection manually. (Optional, as MongoDB creates it on first insert.)

Query:

db.books.insert({

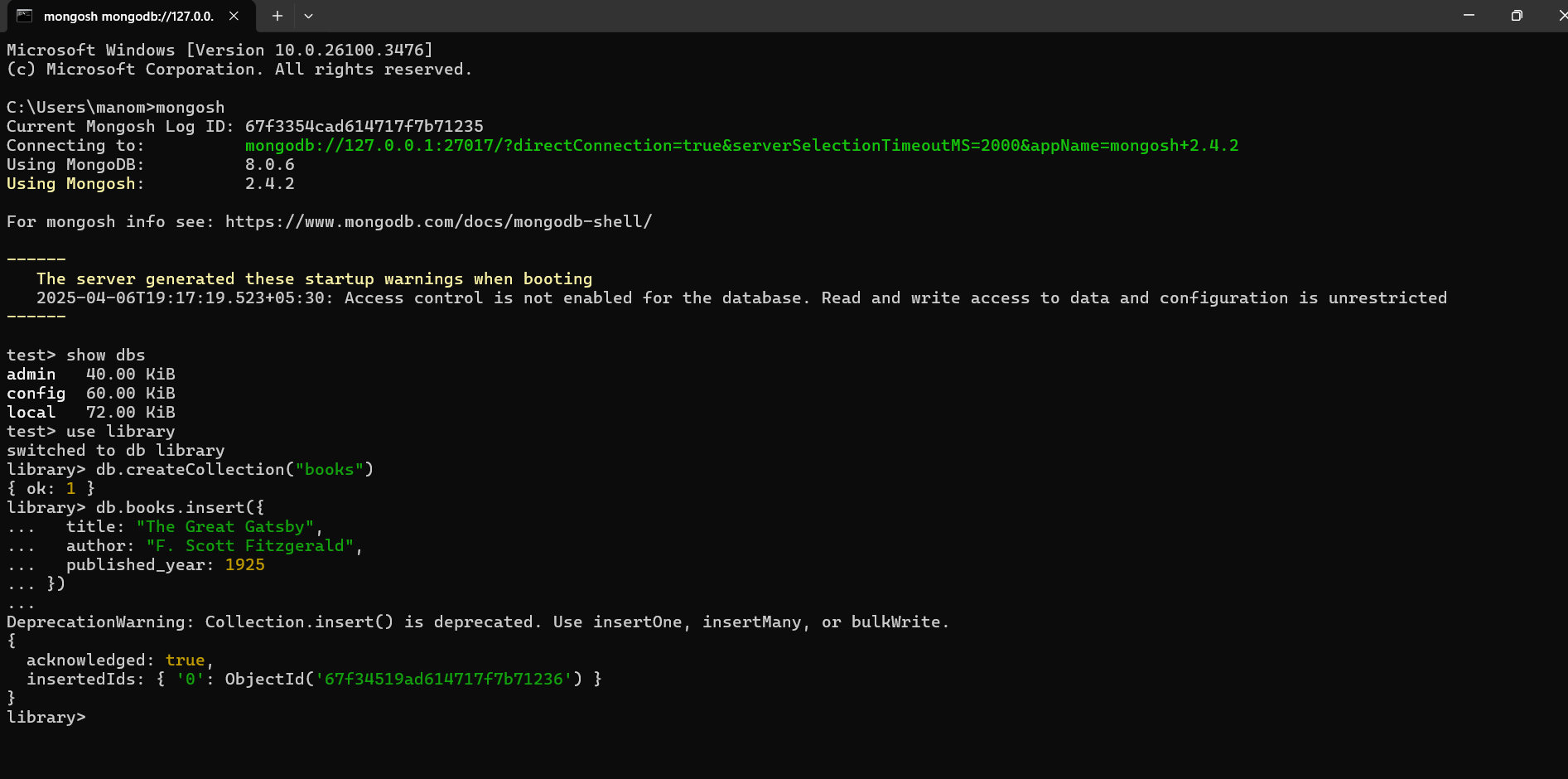
title: "The Great Gatsby",

author: "F. Scott Fitzgerald",

published\_year: 1925

})

**Explanation:** Inserts a document into the collection with title, author, and year.



**2. Read Operation**

Query:

db.books.find()

**Explanation:** Retrieves all documents in the `books` collection.

Query:

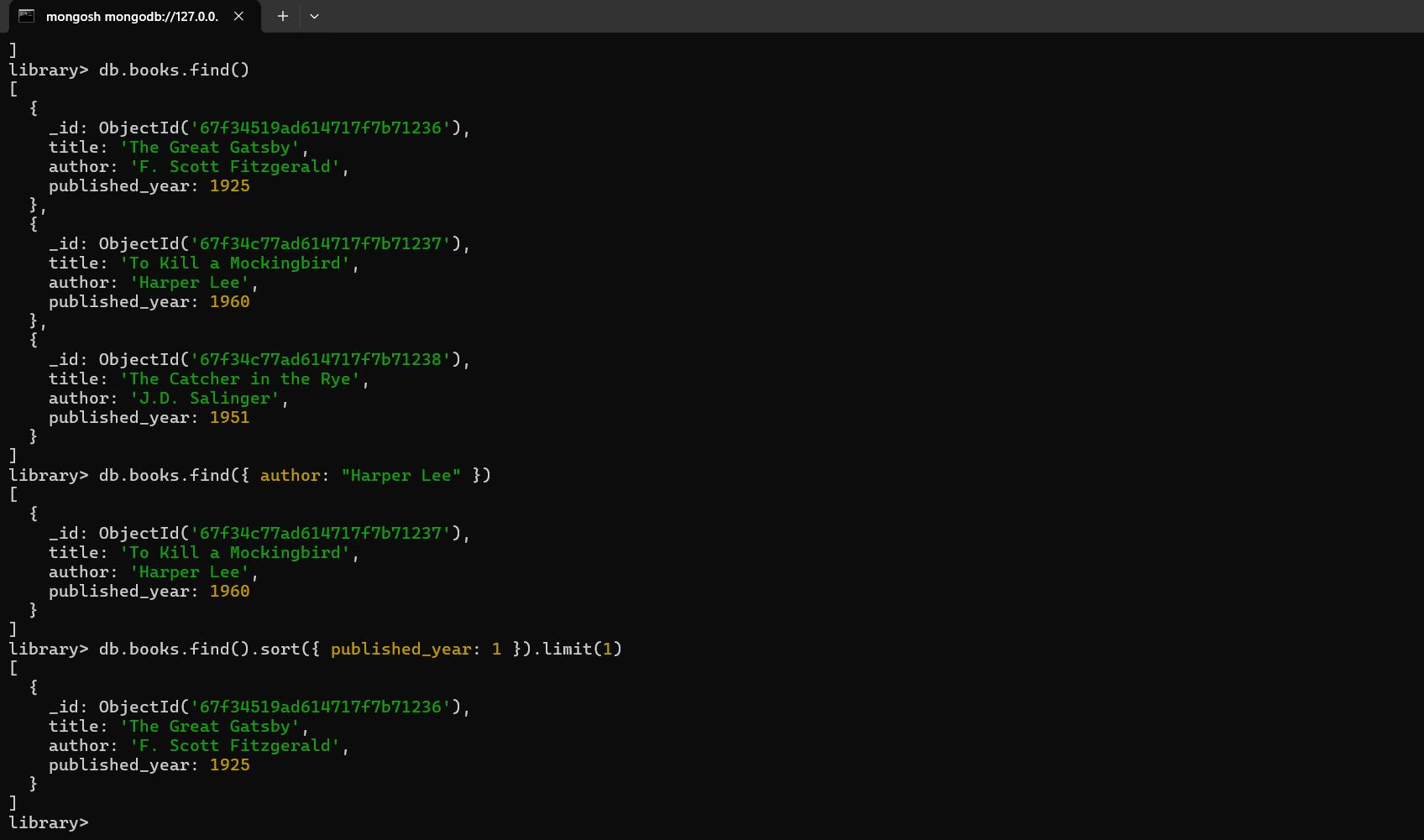
db.books.find({ author: "Harper Lee" })

**Explanation:** Finds all documents where the author is "Harper Lee".

Query:

db.books.find().sort({ published\_year: 1 }).limit(1)

**Explanation:** Sorts all books in ascending order of `published\_year` and shows the earliest one.



**3. Update Operation**

Query:

db.books.update(

{ title: "The Catcher in the Rye" },

{ $set: { published\_year: 2025 } }

)

**Explanation:** Updates the published year of a specific book to the current year.

Query

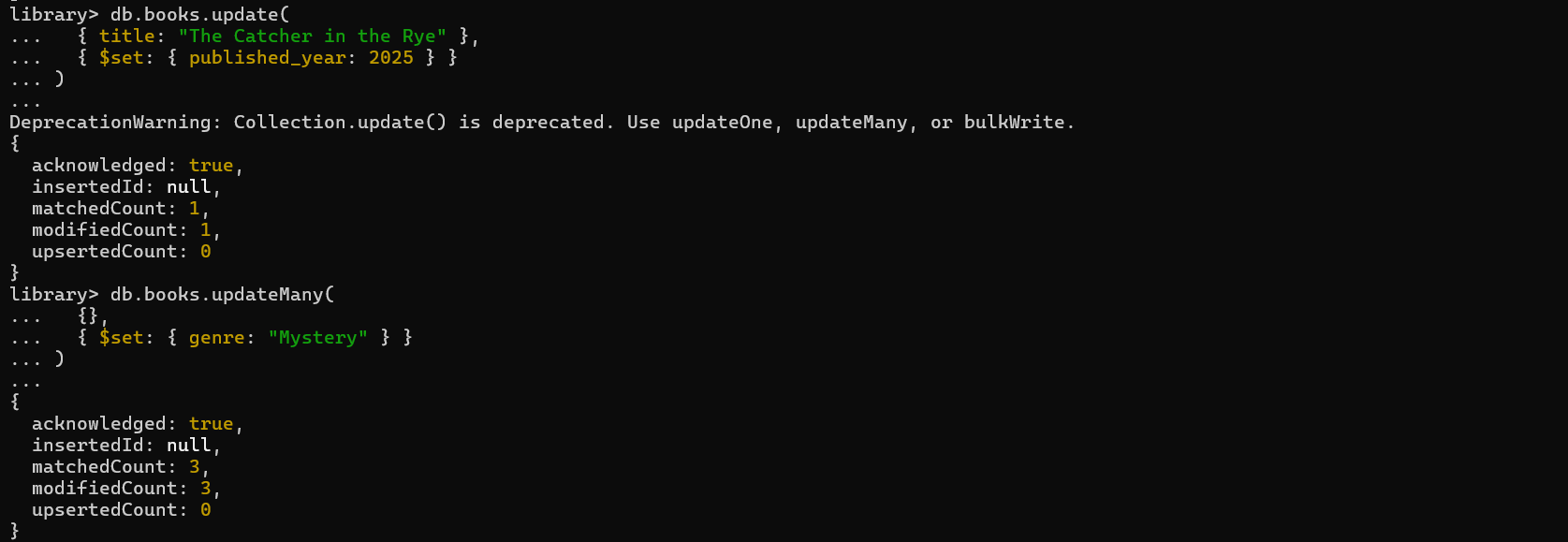
db.books.updateMany(

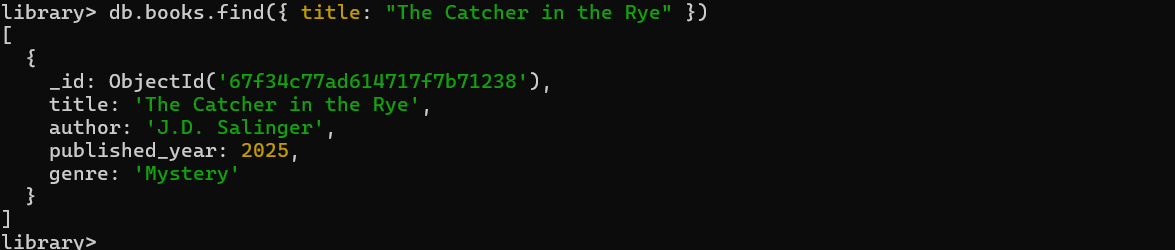
{},

{ $set: { genre: "Mystery" } }

)

**Explanation:** Adds a new field `genre` with value "Mystery" to all books.





**4. Delete Operation**

Query:

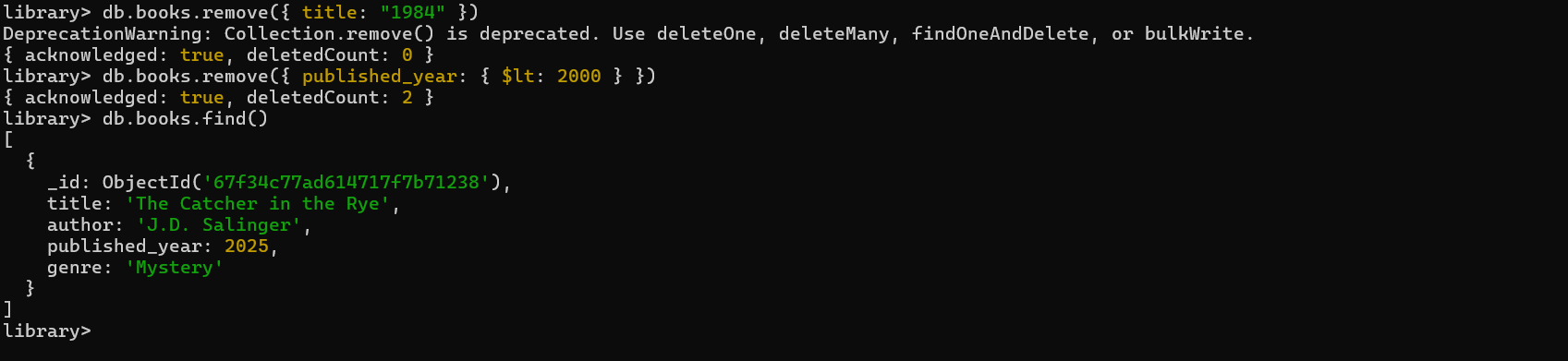
db.books.remove({ title: "1984" })

**Explanation:** Removes the book with title "1984" from the collection.

Query:

db.books.remove({ published\_year: { $lt: 2000 } })

**Explanation:** Deletes all books published before the year 2000.



**5. Advanced Query**

Query:

db.books.find().sort({ published\_year: -1 }).limit(3)

**Explanation:** Finds the top 3 most recently published books using descending sort.

Query:

db.books.find({ title: { $regex: "(MongoDB|NoSQL)", $options: "i" } })

**Explanation:** Finds books with "MongoDB" or "NoSQL" in the title, ignoring case.

Query:

db.books.find({ title: { $regex: "The", $options: "i" } })

**Explanation:**  
Finds books with the word "The" in the title, case-insensitive.

