

Day 7 Coding Assignment — MongoDB Data Modeling and CRUD Operations

User Stories and Tasks User Story 1 —

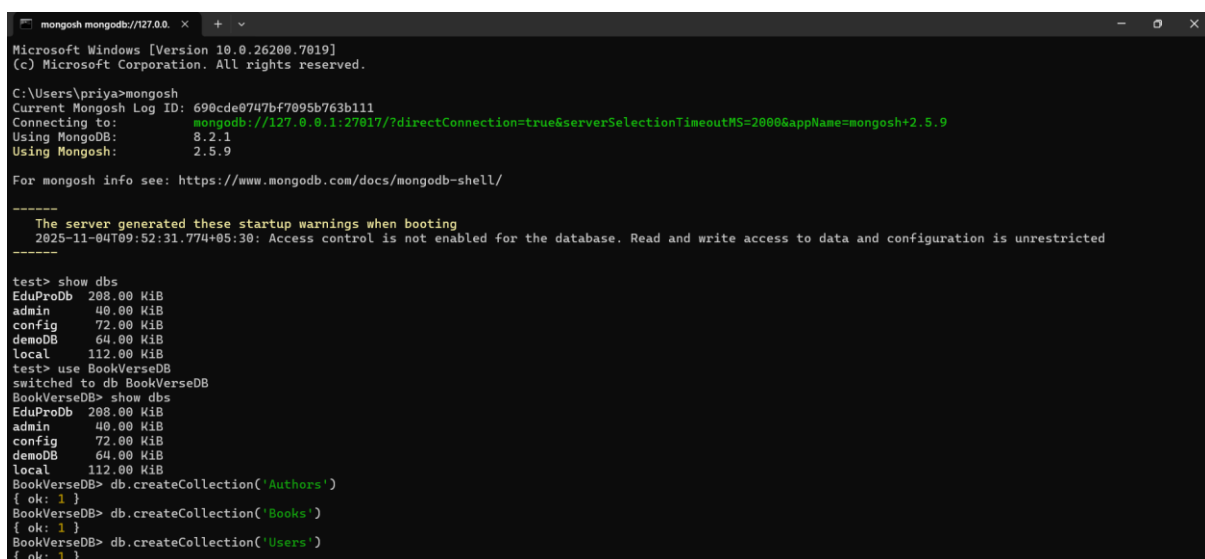
Database Setup and Data Modeling As a developer, I need to design the MongoDB collections to efficiently represent books, authors, and reviews in a NoSQL structure.

Tasks:

1. Create a new MongoDB database named BookVerseDB.
2. Define the following collections and structure them using proper schema modeling principles:
 - Authors: { _id, name, nationality, birthYear }
 - Books: { _id, title, genre, publicationYear, authorId, ratings: [{ user, score, comment }] }
 - Users: { _id, name, email, joinDate }
3. Ensure one-to-many relationships are implemented using references (authorId) and embedded documents (ratings).
4. Insert a few sample documents (at least 3 authors, 5 books, 3 users).

Concepts Covered: NoSQL structure, MongoDB data modelling

Mongosh Screenshots –



```
mongosh mongodb://127.0.0.1:27017
Microsoft Windows [Version 10.0.26200.7019]
(c) Microsoft Corporation. All rights reserved.

C:\Users\priya>mongosh
Current Mongosh Log ID: 690cde0747bf7095b763b111
Connecting to:  mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.5.9
Using MongoDB:  8.2.1
Using Mongosh:  2.5.9

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

-----
The server generated these startup warnings when booting
2025-11-04T09:52:31.774+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
-----

test> show dbs
EduProDb  208.00 KiB
admin     40.00 KiB
config    72.00 KiB
demoDB    64.00 KiB
local     112.00 KiB
test> use BookVerseDB
switched to db BookVerseDB
BookVerseDB> show dbs
EduProDb  208.00 KiB
admin     40.00 KiB
config    72.00 KiB
demoDB    64.00 KiB
local     112.00 KiB
BookVerseDB> db.createCollection('Authors')
{ ok: 1 }
BookVerseDB> db.createCollection('Books')
{ ok: 1 }
BookVerseDB> db.createCollection('Users')
{ ok: 1 }
```

```
mongosh mongodb://127.0.0.1:27017/BookVerseDB
BookVerseDB> db.Authors.insertMany([
...   {
...     name: "Isaac Asimov",
...     nationality: "American",
...     birthYear: 1920
...   },
...   {
...     name: "J.R.R. Tolkien",
...     nationality: "British",
...     birthYear: 1892
...   },
...   {
...     name: "Ursula K. Le Guin",
...     nationality: "American",
...     birthYear: 1929
...   }
... ])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('690ce12947bf7895b763b112'),
    '1': ObjectId('690ce12947bf7895b763b113'),
    '2': ObjectId('690ce12947bf7895b763b114')
  }
}
BookVerseDB> db.Users.insertMany([
...   {
...     name: "Alice Johnson",
...     email: "alice@email.com",
...     joinDate: new Date("2024-08-15")
...   },
...   {
...     name: "Bob Smith",
...     email: "bob@email.com",
...     joinDate: new Date("2024-10-01")
...   },
...   {
...     name: "Carol White",
...     email: "carol@email.com",
...     joinDate: new Date("2024-09-20")
...   }
... ])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('690ce14a47bf7895b763b115'),
    '1': ObjectId('690ce14a47bf7895b763b116'),
    '2': ObjectId('690ce14a47bf7895b763b117')
  }
}
```

```
mongosh mongodb://127.0.0.1:27017/BookVerseDB
BookVerseDB> db.Books.insertMany([
...   {
...     title: "Foundation",
...     genre: "Science Fiction",
...     publicationYear: 1951,
...     authorId: ObjectId('690ce12947bf7895b763b112'),
...     ratings: [
...       { user: "Alice Johnson", score: 5, comment: "Masterpiece of sci-fi" },
...       { user: "Bob Smith", score: 4, comment: "Great worldbuilding" }
...     ]
...   },
...   {
...     title: "I, Robot",
...     genre: "Science Fiction",
...     publicationYear: 1950,
...     authorId: ObjectId('690ce12947bf7895b763b112'),
...     ratings: [
...       { user: "Carol White", score: 5, comment: "Thought-provoking" }
...     ]
...   },
...   {
...     title: "The Hobbit",
...     genre: "Fantasy",
...     publicationYear: 1937,
...     authorId: ObjectId('690ce12947bf7895b763b113'),
...     ratings: [
...       { user: "Alice Johnson", score: 5, comment: "Timeless classic" },
...       { user: "Bob Smith", score: 5, comment: "Amazing adventure" }
...     ]
...   },
...   {
...     title: "The Lord of the Rings",
...     genre: "Fantasy",
...     publicationYear: 1954,
...     authorId: ObjectId('690ce12947bf7895b763b113'),
...     ratings: [
...       { user: "Carol White", score: 5, comment: "Epic fantasy" }
...     ]
...   },
...   {
...     title: "The Left Hand of Darkness",
...     genre: "Science Fiction",
...     publicationYear: 2017,
...     authorId: ObjectId('690ce14a47bf7895b763b114'),
...     ratings: [
...     ]
...   }
... ])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('690ce14a47bf7895b763b118'),
    '1': ObjectId('690ce14a47bf7895b763b119'),
    '2': ObjectId('690ce14a47bf7895b763b120'),
    '3': ObjectId('690ce14a47bf7895b763b121'),
    '4': ObjectId('690ce14a47bf7895b763b122')
  }
}
```

```
mongosh mongodb://127.0.0.1:27017/BookVerseDB> use BookVerseDB
switched to db BookVerseDB
BookVerseDB> insert({
...   { user: "Bob Smith", score: 5, comment: "Amazing adventure" }
... },
... {
...   title: "The Lord of the Rings",
...   genre: "Fantasy",
...   publicationYear: 1984,
...   authorId: ObjectId("690ccec2c4b373d14a63b113"),
...   ratings: [
...     { user: "Carol White", score: 5, comment: "Epic fantasy" }
...   ]
... },
... {
...   title: "The Left Hand of Darkness",
...   genre: "Science Fiction",
...   publicationYear: 2017,
...   authorId: ObjectId("690ccec2c4b373d14a63b114"),
...   ratings: [
...     { user: "Alice Johnson", score: 4, comment: "Unique perspective" },
...     { user: "Bob Smith", score: 4, comment: "Interesting concepts" }
...   ]
... })
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('690ce16b47bf7095b763b118'),
    '1': ObjectId('690ce16b47bf7095b763b119'),
    '2': ObjectId('690ce16b47bf7095b763b11a'),
    '3': ObjectId('690ce16b47bf7095b763b11b'),
    '4': ObjectId('690ce16b47bf7095b763b11c')
  }
}
BookVerseDB> |
```

User Story 2 —

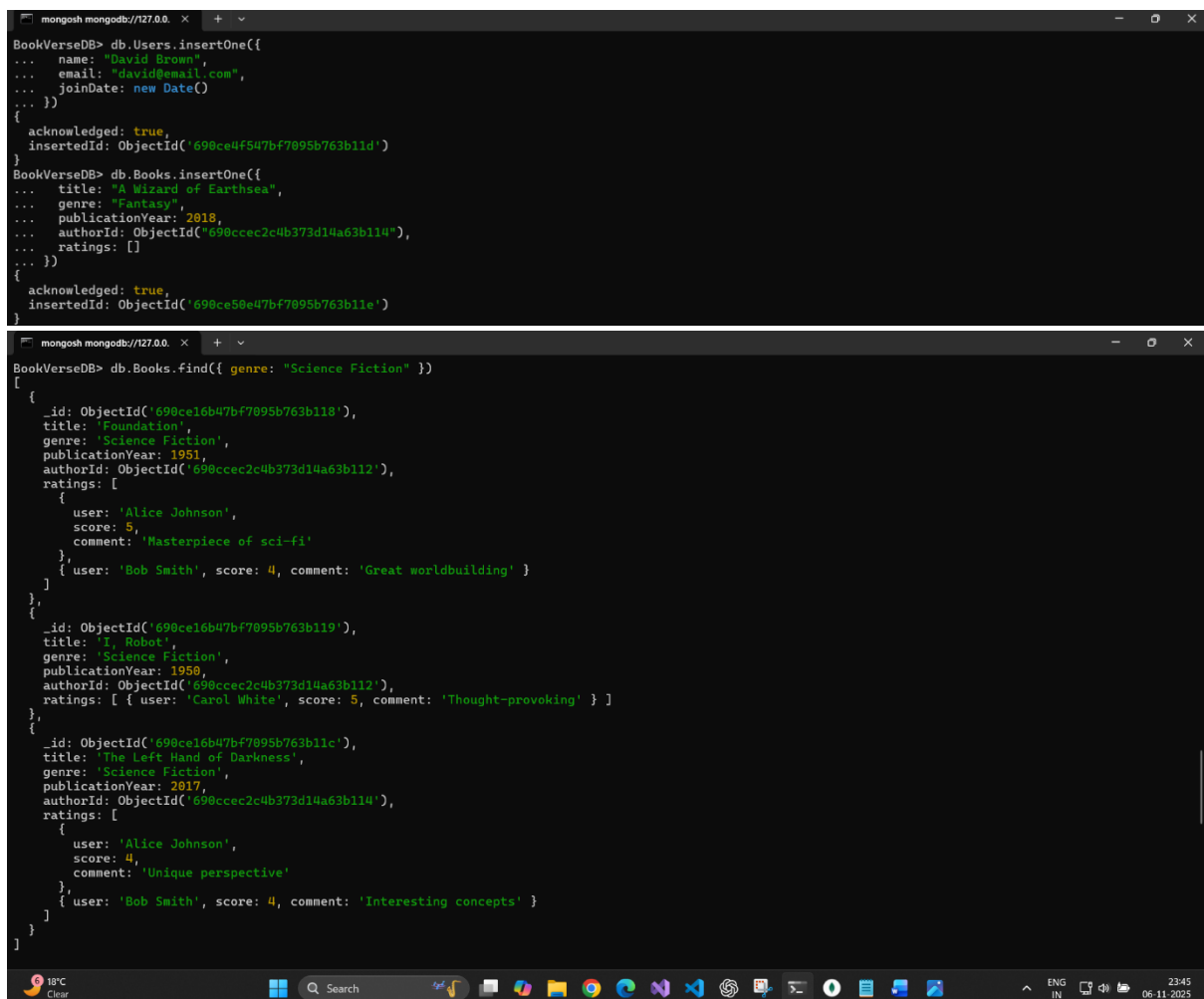
CRUD Operations As a backend developer, I need to perform basic Create, Read, Update, and Delete operations on the collections.

Tasks:

1. Insert new users and books into the database.
2. Retrieve all books of the genre “Science Fiction”.
3. Update the publicationYear of one book.
4. Delete one user record from the collection.
5. Add a new rating to a book document using the \$push operator.

Concepts Covered: CRUD operations, update modifiers

Mongosh Screenshots –



The image shows two screenshots of the mongosh terminal interface. The top screenshot shows the insertion of a new user and a new book into the database. The bottom screenshot shows the retrieval of all books of the genre "Science Fiction".

```
BookVerseDB> db.Users.insertOne({
...   name: "David Brown",
...   email: "david@email.com",
...   joinDate: new Date()
... })
{
  acknowledged: true,
  insertedId: ObjectId('690ce4f547bf7895b763b11d')
}
BookVerseDB> db.Books.insertOne({
...   title: "A Wizard of Earthsea",
...   genre: "Fantasy",
...   publicationYear: 2018,
...   authorId: ObjectId('690ccec2c4b373d14a63b114'),
...   ratings: []
... })
{
  acknowledged: true,
  insertedId: ObjectId('690ce58e47bf7895b763b11e')
}
```

```
BookVerseDB> db.Books.find({ genre: "Science Fiction" })
[
  {
    _id: ObjectId('690ce16b47bf7895b763b118'),
    title: 'Foundation',
    genre: 'Science Fiction',
    publicationYear: 1951,
    authorId: ObjectId('690ccec2c4b373d14a63b112'),
    ratings: [
      {
        user: 'Alice Johnson',
        score: 5,
        comment: 'Masterpiece of sci-fi'
      },
      { user: 'Bob Smith', score: 4, comment: 'Great worldbuilding' }
    ]
  },
  {
    _id: ObjectId('690ce16b47bf7895b763b119'),
    title: 'I, Robot',
    genre: 'Science Fiction',
    publicationYear: 1950,
    authorId: ObjectId('690ccec2c4b373d14a63b112'),
    ratings: [ { user: 'Carol White', score: 5, comment: 'Thought-provoking' } ]
  },
  {
    _id: ObjectId('690ce16b47bf7895b763b11c'),
    title: 'The Left Hand of Darkness',
    genre: 'Science Fiction',
    publicationYear: 2017,
    authorId: ObjectId('690ccec2c4b373d14a63b114'),
    ratings: [
      {
        user: 'Alice Johnson',
        score: 4,
        comment: 'Unique perspective'
      },
      { user: 'Bob Smith', score: 4, comment: 'Interesting concepts' }
    ]
  }
]
```

```
mongosh mongodb://127.0.0.1:27020
BookVerseDB> db.Books.updateOne(
...   { title: "The Left Hand of Darkness" },
...   { $set: { publicationYear: 1969 } }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
BookVerseDB> db.Books.findOne({ title: "The Left Hand of Darkness" })
{
  _id: ObjectId('690ce16b47bf7095b763b11c'),
  title: 'The Left Hand of Darkness',
  genre: 'Science Fiction',
  publicationYear: 1969,
  authorId: ObjectId('690ccce2c4b373d14a63b114'),
  ratings: [
    { user: 'Alice Johnson', score: 4, comment: 'Unique perspective' },
    { user: 'Bob Smith', score: 4, comment: 'Interesting concepts' }
  ]
}
BookVerseDB> db.Users.deleteOne({ name: "David Brown" })
{ acknowledged: true, deletedCount: 1 }
BookVerseDB> db.Users.find({ name: "David Brown" })
[]
BookVerseDB> db.Books.updateOne(
...   { title: "Foundation" },
...   {
...     $push: {
...       ratings: {
...         user: "Carol White",
...         score: 5,
...         comment: "Best sci-fi series ever"
...       }
...     }
...   }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
BookVerseDB> db.Books.findOne({ title: "Foundation" })
{
  _id: ObjectId('690ce16b47bf7095b763b11d'),
  title: 'Foundation',
  genre: 'Science Fiction',
  publicationYear: 1951,
  authorId: ObjectId('690ccce2c4b373d14a63b112'),
  ratings: [
    { user: 'Alice Johnson', score: 5, comment: 'Masterpiece of sci-fi' },
    { user: 'Bob Smith', score: 4, comment: 'Great worldbuilding' },
    { user: 'Carol White', score: 5, comment: 'Best sci-fi series ever' }
  ]
}
BookVerseDB>
```

User Story 3 —

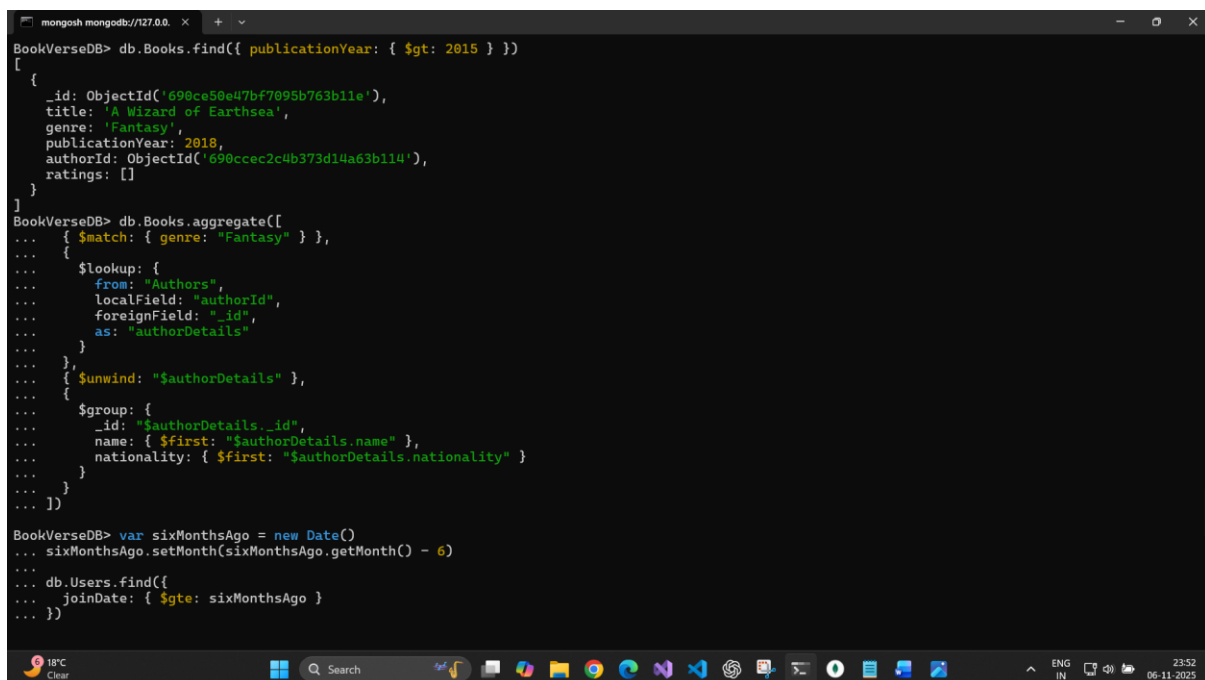
Querying and Filtering Data As an application developer, I need to query data for the web interface based on filters.

Tasks:

1. Retrieve all books published after 2015.
2. Find authors who have written books in the “Fantasy” genre.
3. Retrieve all users who joined within the last 6 months.
4. Find books with an average rating greater than 4.

Concepts Covered: Query filters, conditional operators, nested field access

Mongosh Screenshots –



```
mongosh mongodb://127.0.0.1:27020
BookVerseDB> db.Books.find({ publicationYear: { $gt: 2015 } })
[
  {
    _id: ObjectId('690ce50e47bf7095b763b11e'),
    title: 'A Wizard of Earthsea',
    genre: 'Fantasy',
    publicationYear: 2018,
    authorId: ObjectId('690ccce2c4b373d14a63b114'),
    ratings: []
  }
]
BookVerseDB> db.Books.aggregate([
...   { $match: { genre: "Fantasy" } },
...   {
...     $lookup: {
...       from: "Authors",
...       localField: "authorId",
...       foreignField: "_id",
...       as: "authorDetails"
...     }
...   },
...   { $unwind: "$authorDetails" },
...   {
...     $group: {
...       _id: "$authorDetails._id",
...       name: { $first: "$authorDetails.name" },
...       nationality: { $first: "$authorDetails.nationality" }
...     }
...   }
... ])
BookVerseDB> var sixMonthsAgo = new Date()
... sixMonthsAgo.setMonth(sixMonthsAgo.getMonth() - 6)
...
... db.Users.find({
...   joinDate: { $gte: sixMonthsAgo }
... })
```

```
mongosh mongodb://127.0.0.1:27000
BookVerseDB> db.Books.aggregate([
...   {
...     $addFields: {
...       avgRating: { $avg: "$ratings.score" }
...     },
...   },
...   {
...     $match: {
...       avgRating: { $gt: 4 }
...     }
...   },
...   {
...     $project: {
...       title: 1,
...       genre: 1,
...       avgRating: 1,
...       ratings: 1
...     }
...   }
... ])
[
  {
    _id: ObjectId('690ce16b47bf7095b763b118'),
    title: 'Foundation',
    genre: 'Science Fiction',
    ratings: [
      {
        user: 'Alice Johnson',
        score: 5,
        comment: 'Masterpiece of sci-fi'
      },
      {
        user: 'Bob Smith', score: 4, comment: 'Great worldbuilding' },
      {
        user: 'Carol White',
        score: 5,
        comment: 'Best sci-fi series ever'
      }
    ]
  },
  {
    _id: ObjectId('690ce16b47bf7095b763b119'),
    title: 'I, Robot',
    genre: 'Science Fiction',
    ratings: [ { user: 'Carol White', score: 5, comment: 'Thought-provoking' } ],
    avgRating: 5
  },
  {
    _id: ObjectId('690ce16b47bf7095b763b11a'),
    title: 'The Hobbit',
    genre: 'Fantasy',
    ratings: [
      { user: 'Alice Johnson', score: 5, comment: 'Timeless classic' },
      { user: 'Bob Smith', score: 5, comment: 'Amazing adventure' }
    ],
    avgRating: 5
  },
  {
    _id: ObjectId('690ce16b47bf7095b763b11b'),
    title: 'The Lord of the Rings',
    genre: 'Fantasy',
    ratings: [ { user: 'Carol White', score: 5, comment: 'Epic fantasy' } ],
    avgRating: 5
  }
]
BookVerseDB>
```