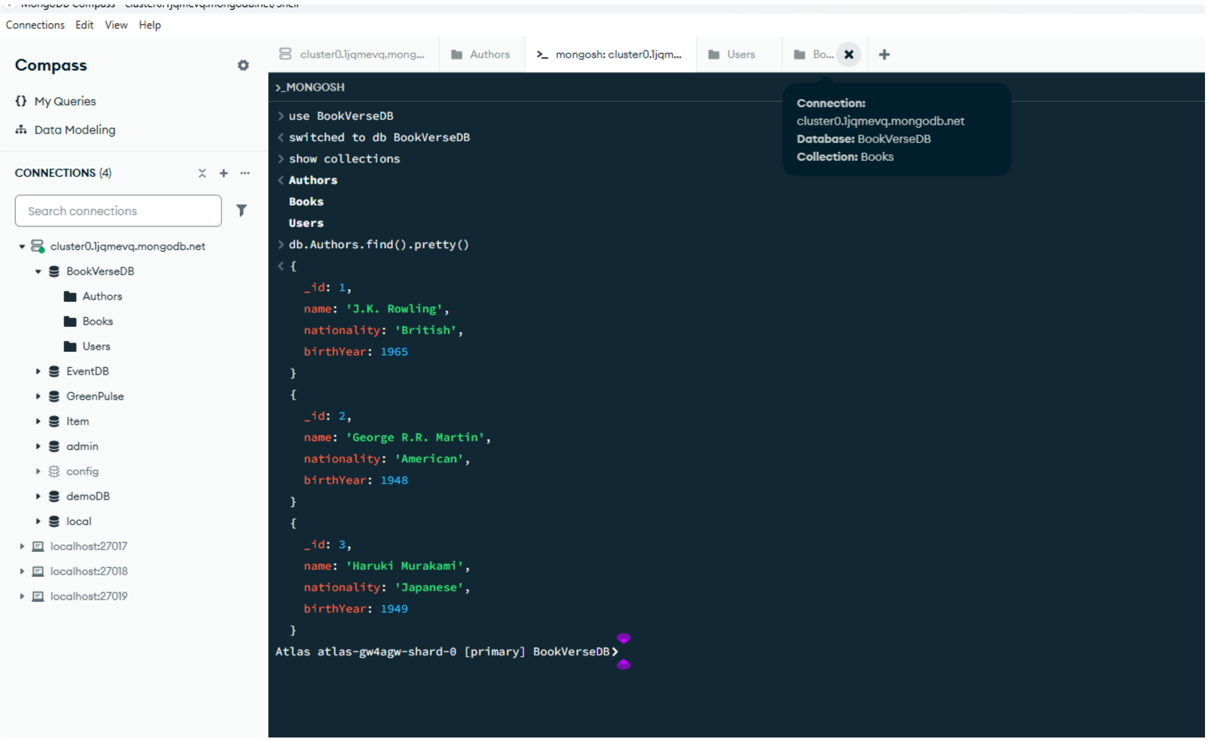
Day 7 coding assessment

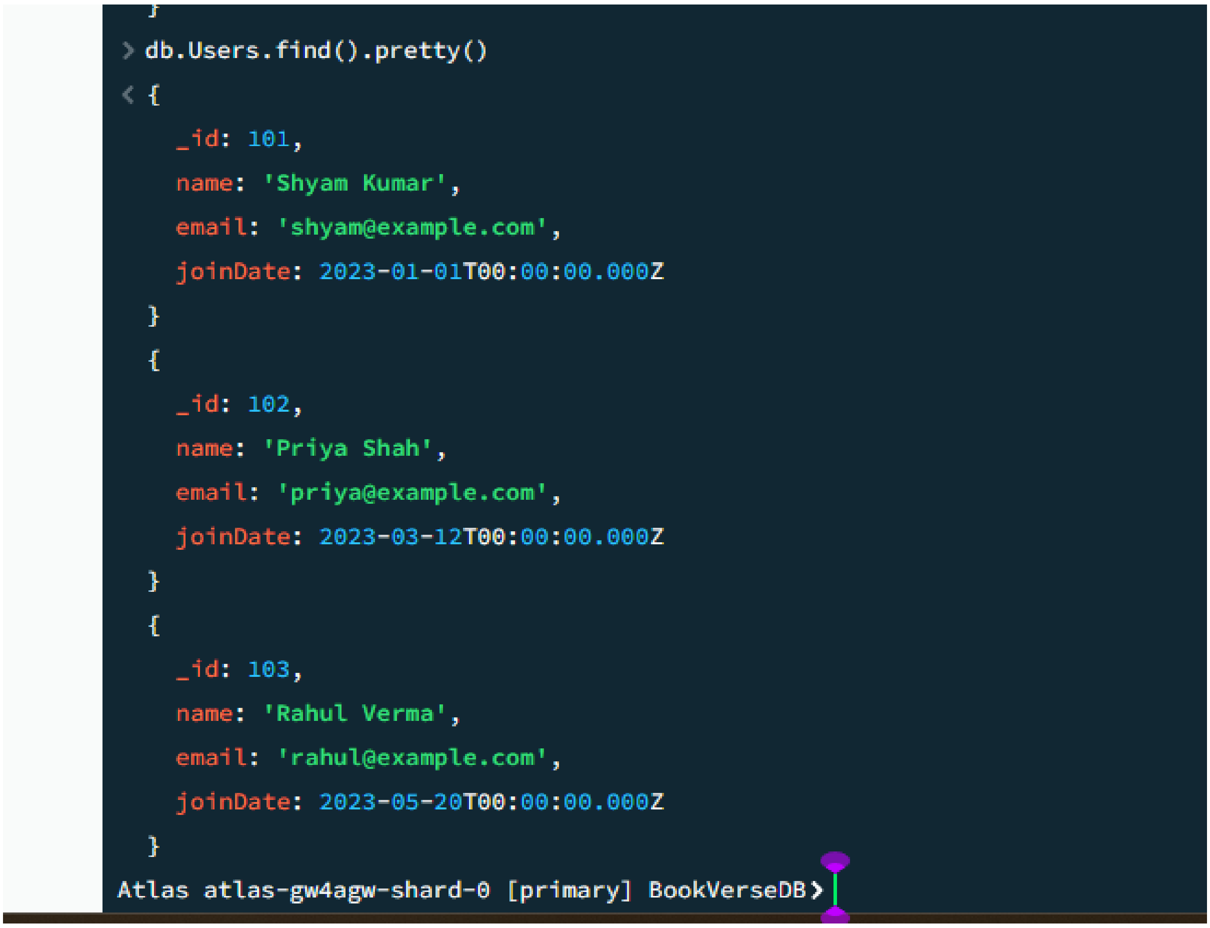
User story 1 output:



Connection: cluster0.1jqmevq.mongodb.net
Database: BookVerseDB
Collection: Books

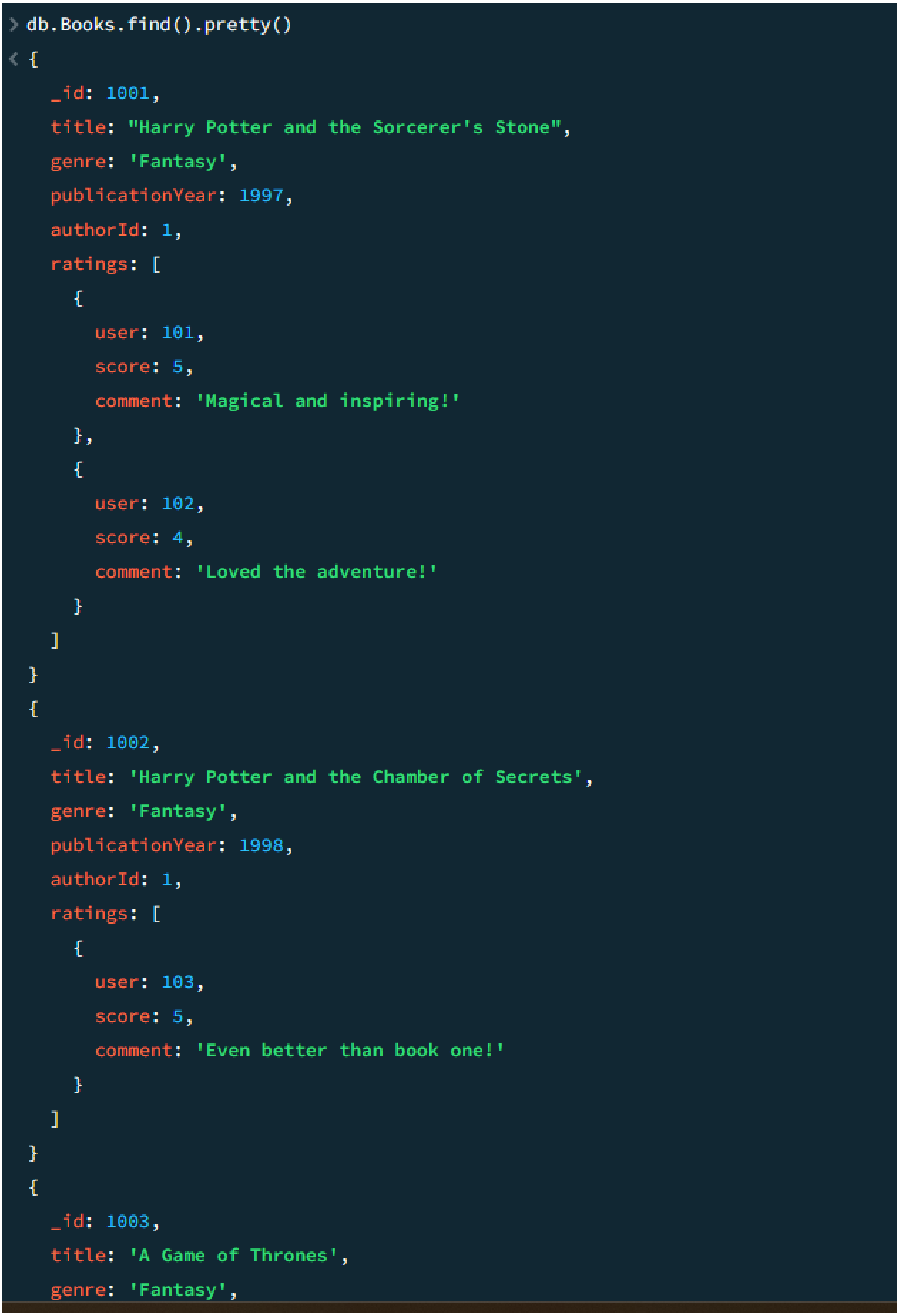
```
> use BookVerseDB
< switched to db BookVerseDB
> show collections
< Authors
  Books
  Users
> db.Authors.find().pretty()
< {
    _id: 1,
    name: 'J.K. Rowling',
    nationality: 'British',
    birthYear: 1965
  }
  {
    _id: 2,
    name: 'George R.R. Martin',
    nationality: 'American',
    birthYear: 1948
  }
  {
    _id: 3,
    name: 'Haruki Murakami',
    nationality: 'Japanese',
    birthYear: 1949
  }
Atlas atlas-gw4agw-shard-0 [primary] BookVerseDB>
```

This screenshot output displays the database “BookVerseDB” is created with collections “Authors”, “Users” and “Books”. It also displays the data in “Authors” collection

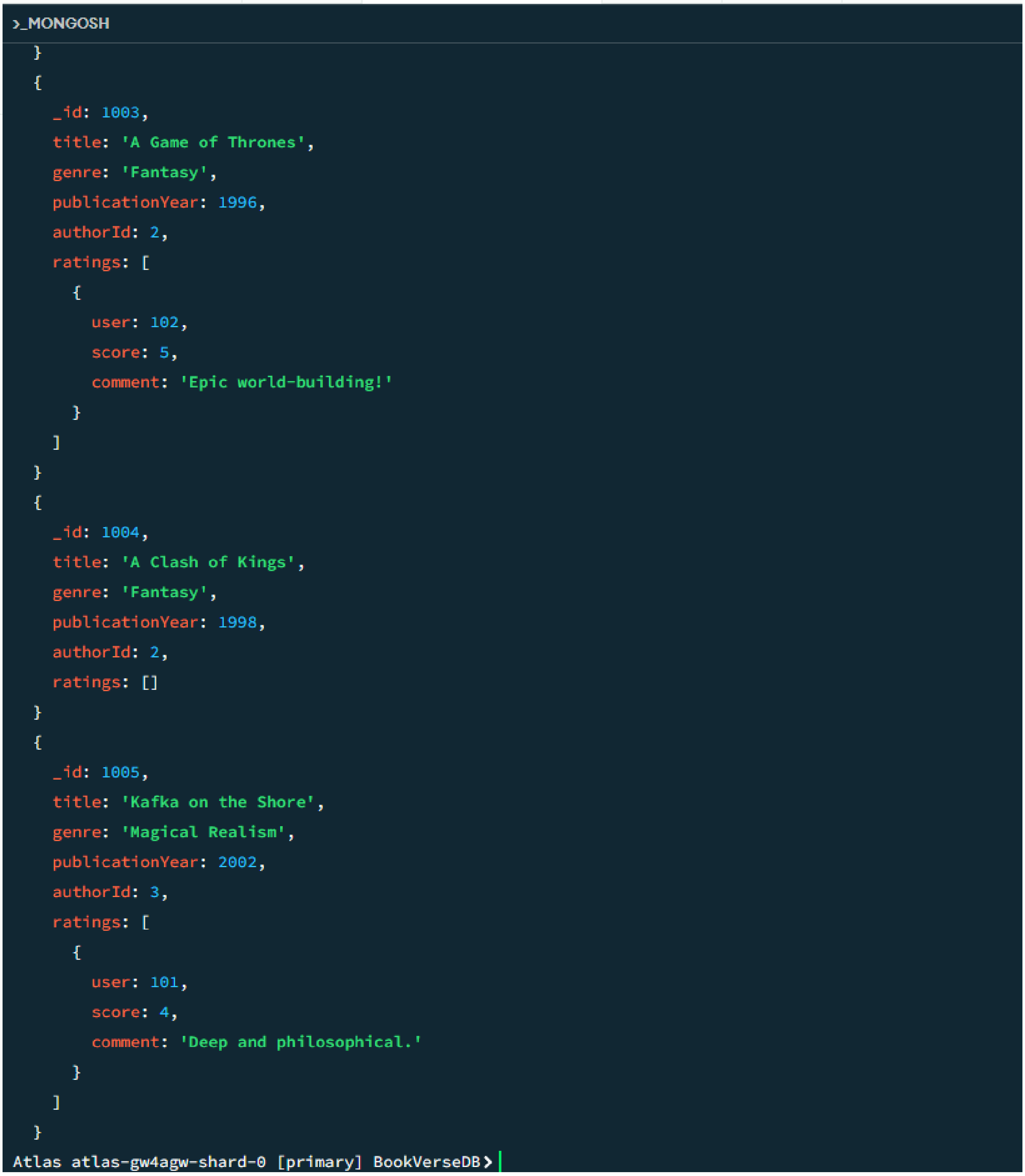


```
  }
> db.Users.find().pretty()
< {
    _id: 101,
    name: 'Shyam Kumar',
    email: 'shyam@example.com',
    joinDate: 2023-01-01T00:00:00.000Z
  }
  {
    _id: 102,
    name: 'Priya Shah',
    email: 'priya@example.com',
    joinDate: 2023-03-12T00:00:00.000Z
  }
  {
    _id: 103,
    name: 'Rahul Verma',
    email: 'rahul@example.com',
    joinDate: 2023-05-20T00:00:00.000Z
  }
Atlas atlas-gw4agw-shard-0 [primary] BookVerseDB>
```

This screenshot output displays “Users” collection.



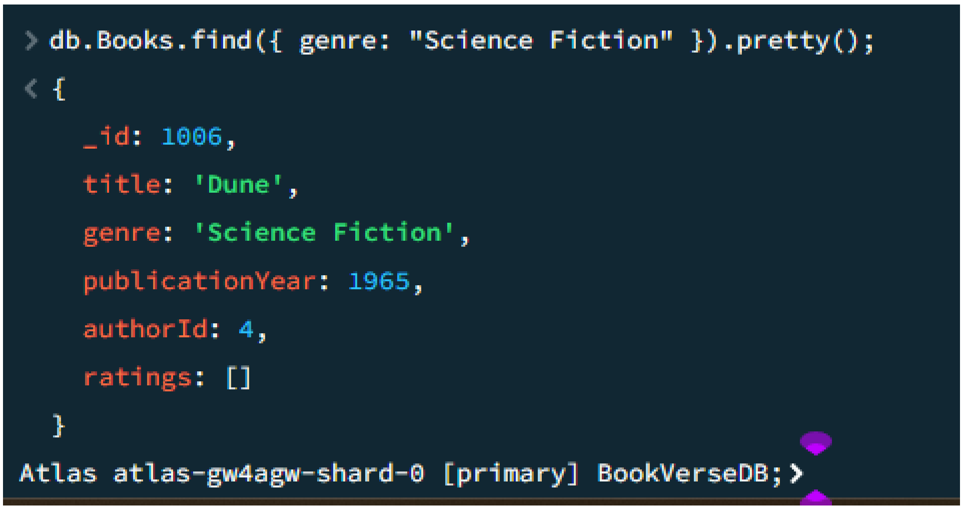
```
> db.Books.find().pretty()
< {
    _id: 1001,
    title: "Harry Potter and the Sorcerer's Stone",
    genre: 'Fantasy',
    publicationYear: 1997,
    authorId: 1,
    ratings: [
      {
        user: 101,
        score: 5,
        comment: 'Magical and inspiring!'
      },
      {
        user: 102,
        score: 4,
        comment: 'Loved the adventure!'
      }
    ]
  }
  {
    _id: 1002,
    title: 'Harry Potter and the Chamber of Secrets',
    genre: 'Fantasy',
    publicationYear: 1998,
    authorId: 1,
    ratings: [
      {
        user: 103,
        score: 5,
        comment: 'Even better than book one!'
      }
    ]
  }
  {
    _id: 1003,
    title: 'A Game of Thrones',
    genre: 'Fantasy',
```



```
  }
  {
    _id: 1003,
    title: 'A Game of Thrones',
    genre: 'Fantasy',
    publicationYear: 1996,
    authorId: 2,
    ratings: [
      {
        user: 102,
        score: 5,
        comment: 'Epic world-building!'
      }
    ]
  }
  {
    _id: 1004,
    title: 'A Clash of Kings',
    genre: 'Fantasy',
    publicationYear: 1998,
    authorId: 2,
    ratings: []
  }
  {
    _id: 1005,
    title: 'Kafka on the Shore',
    genre: 'Magical Realism',
    publicationYear: 2002,
    authorId: 3,
    ratings: [
      {
        user: 101,
        score: 4,
        comment: 'Deep and philosophical.'
      }
    ]
  }
Atlas atlas-gw4agw-shard-0 [primary] BookVerseDB>
```

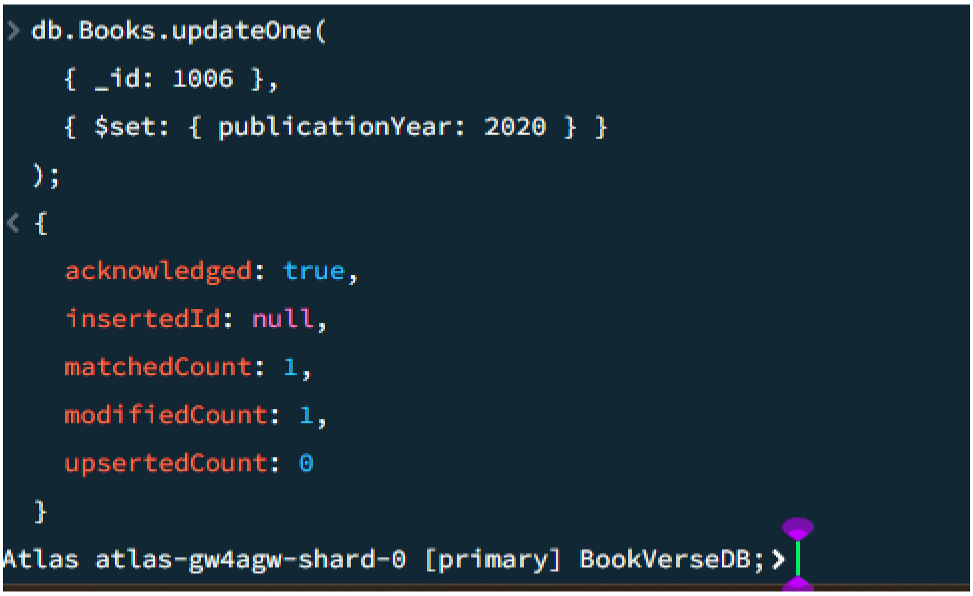
These two screenshots display the data in “Books” collection.

User story 2 output:



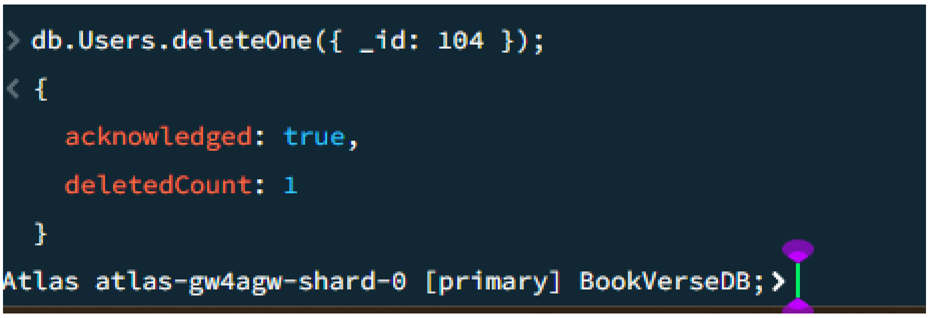
```
> db.Books.find({ genre: "Science Fiction" }).pretty();
< {
    _id: 1006,
    title: 'Dune',
    genre: 'Science Fiction',
    publicationYear: 1965,
    authorId: 4,
    ratings: []
  }
Atlas atlas-gw4agw-shard-0 [primary] BookVerseDB;>
```

New book is inserted into the "Books" collection and it is retrieving books of the genre 'Science Fiction'.



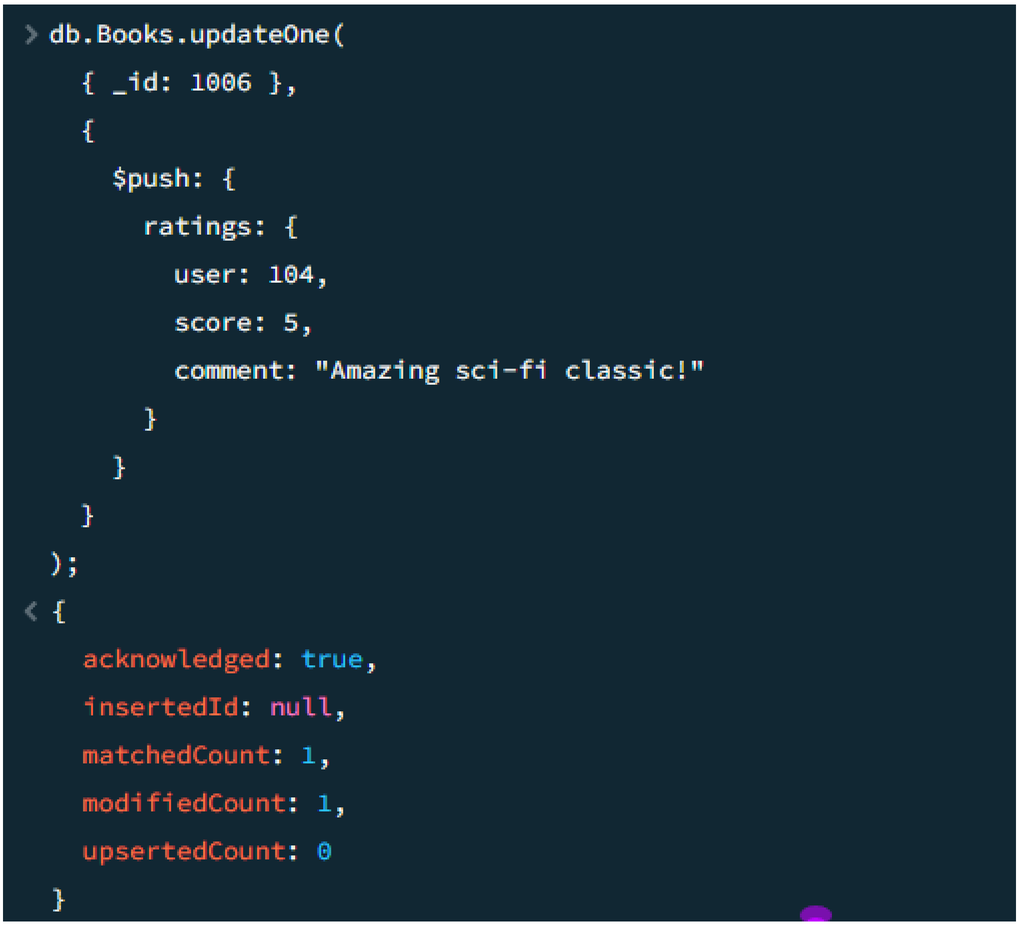
```
> db.Books.updateOne(
    { _id: 1006 },
    { $set: { publicationYear: 2020 } }
  );
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 1,
    modifiedCount: 1,
    upsertedCount: 0
  }
Atlas atlas-gw4agw-shard-0 [primary] BookVerseDB;>
```

This output means the query successfully found the document (matchedCount: 1), but changes were made because the value was updated from (modifiedCount: 0) to (modifiedCount: 1).



```
> db.Users.deleteOne({ _id: 104 });
< {
    acknowledged: true,
    deletedCount: 1
  }
Atlas atlas-gw4agw-shard-0 [primary] BookVerseDB;>
```

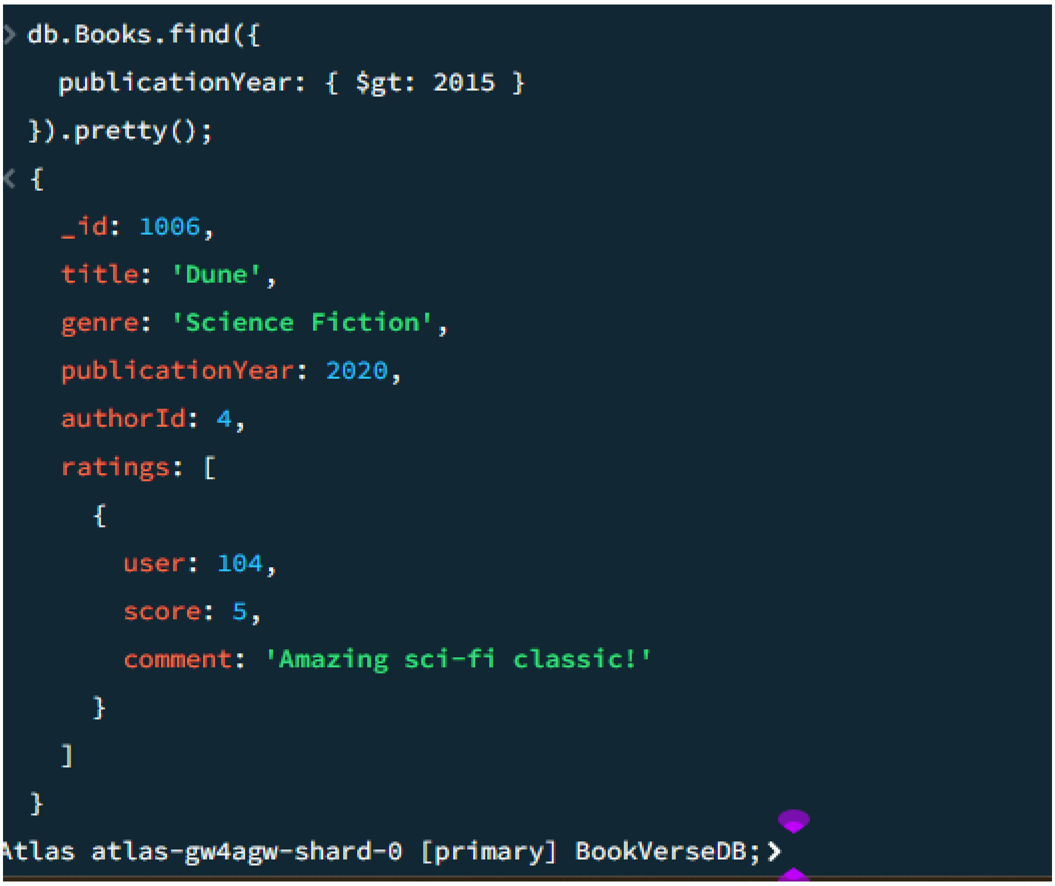
The User with id: 104 was deleted and the count was updated.



```
> db.Books.updateOne(
    { _id: 1006 },
    {
      $push: {
        ratings: {
          user: 104,
          score: 5,
          comment: "Amazing sci-fi classic!"
        }
      }
    }
  );
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 1,
    modifiedCount: 1,
    upsertedCount: 0
  }
```

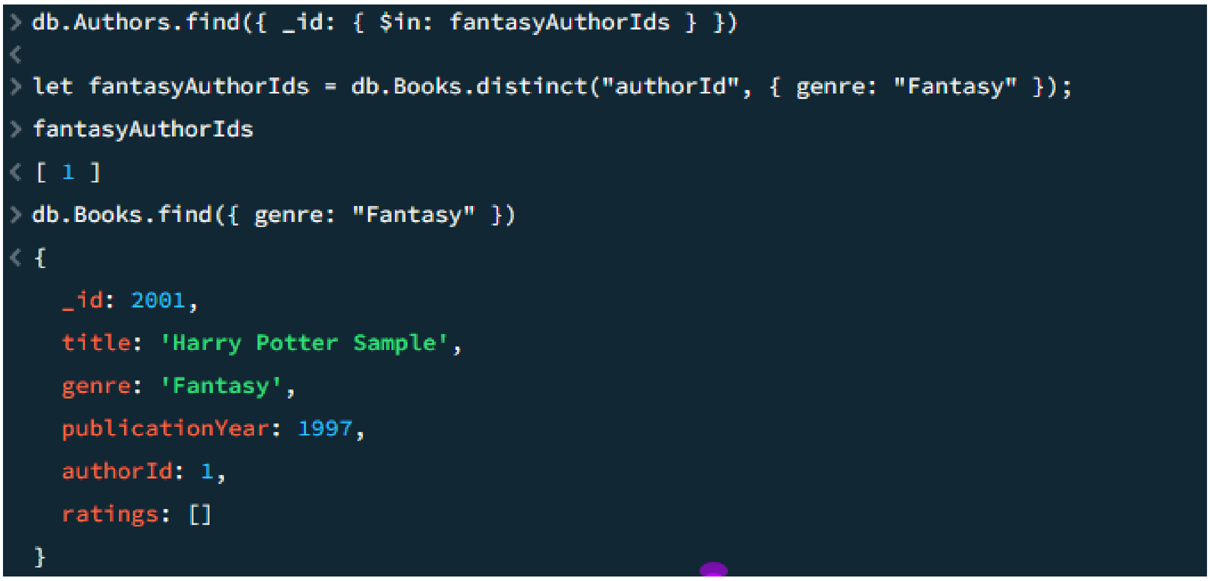
This output shows we successfully found out the book (matchedCount: 1) and added new rating, updating the document (modifiedCount: 1). No new record was created (upsertedCount: 0).

User story 3 output



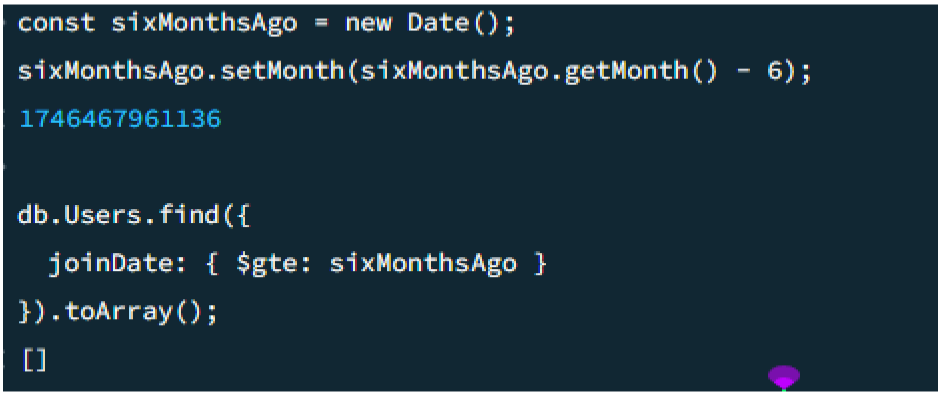
```
> db.Books.find({
    publicationYear: { $gt: 2015 }
  }).pretty();
< {
    _id: 1006,
    title: 'Dune',
    genre: 'Science Fiction',
    publicationYear: 2020,
    authorId: 4,
    ratings: [
      {
        user: 104,
        score: 5,
        comment: 'Amazing sci-fi classic!'
      }
    ]
  }
Atlas atlas-gw4agw-shard-0 [primary] BookVerseDB;>
```

The books published after 2015 is displayed.



```
> db.Authors.find({ _id: { $in: fantasyAuthorIds } })
<
> let fantasyAuthorIds = db.Books.distinct("authorId", { genre: "Fantasy" });
> fantasyAuthorIds
< [ 1 ]
> db.Books.find({ genre: "Fantasy" })
< {
    _id: 2001,
    title: 'Harry Potter Sample',
    genre: 'Fantasy',
    publicationYear: 1997,
    authorId: 1,
    ratings: []
  }
```

The book with fantasy genre



```
const sixMonthsAgo = new Date();
sixMonthsAgo.setMonth(sixMonthsAgo.getMonth() - 6);
1746467961136

db.Users.find({
  joinDate: { $gte: sixMonthsAgo }
}).toArray();
[]
```

No user joined in the last six months.