Coding Assignment — MongoDB Data Modeling and CRUD Operations

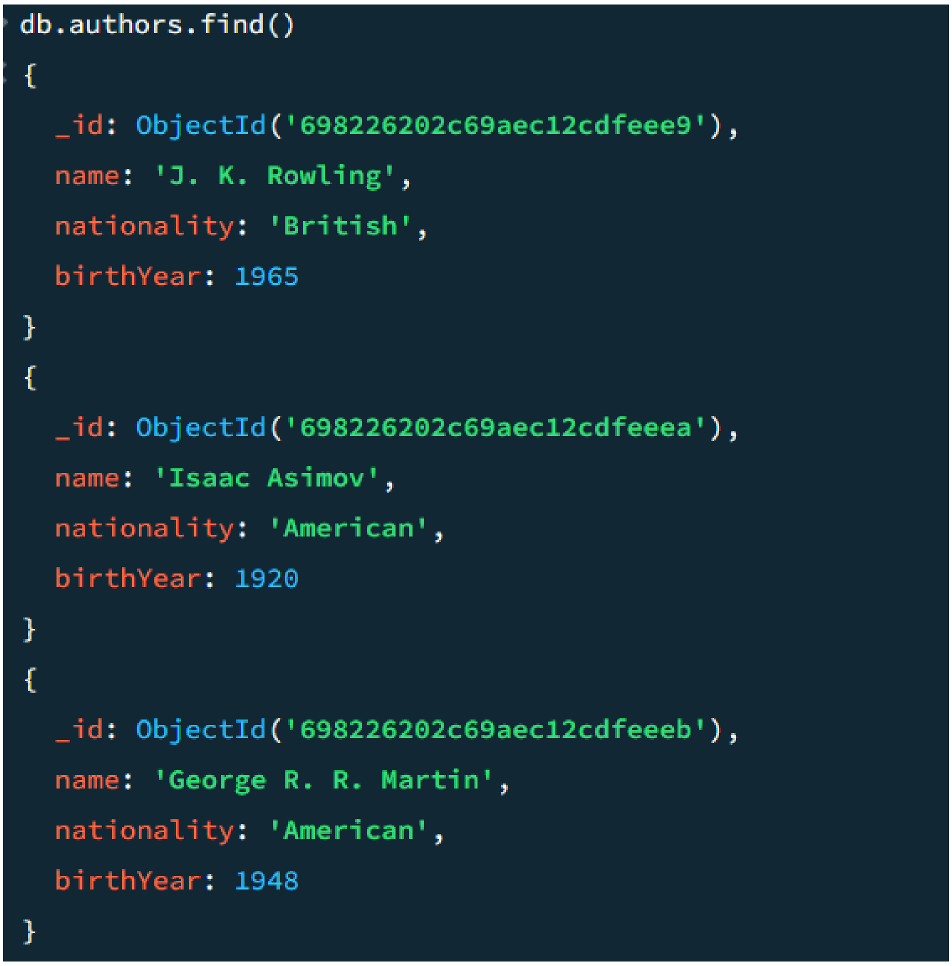
User Story 1 — Database Setup and Data Modeling

use BookVerseDB

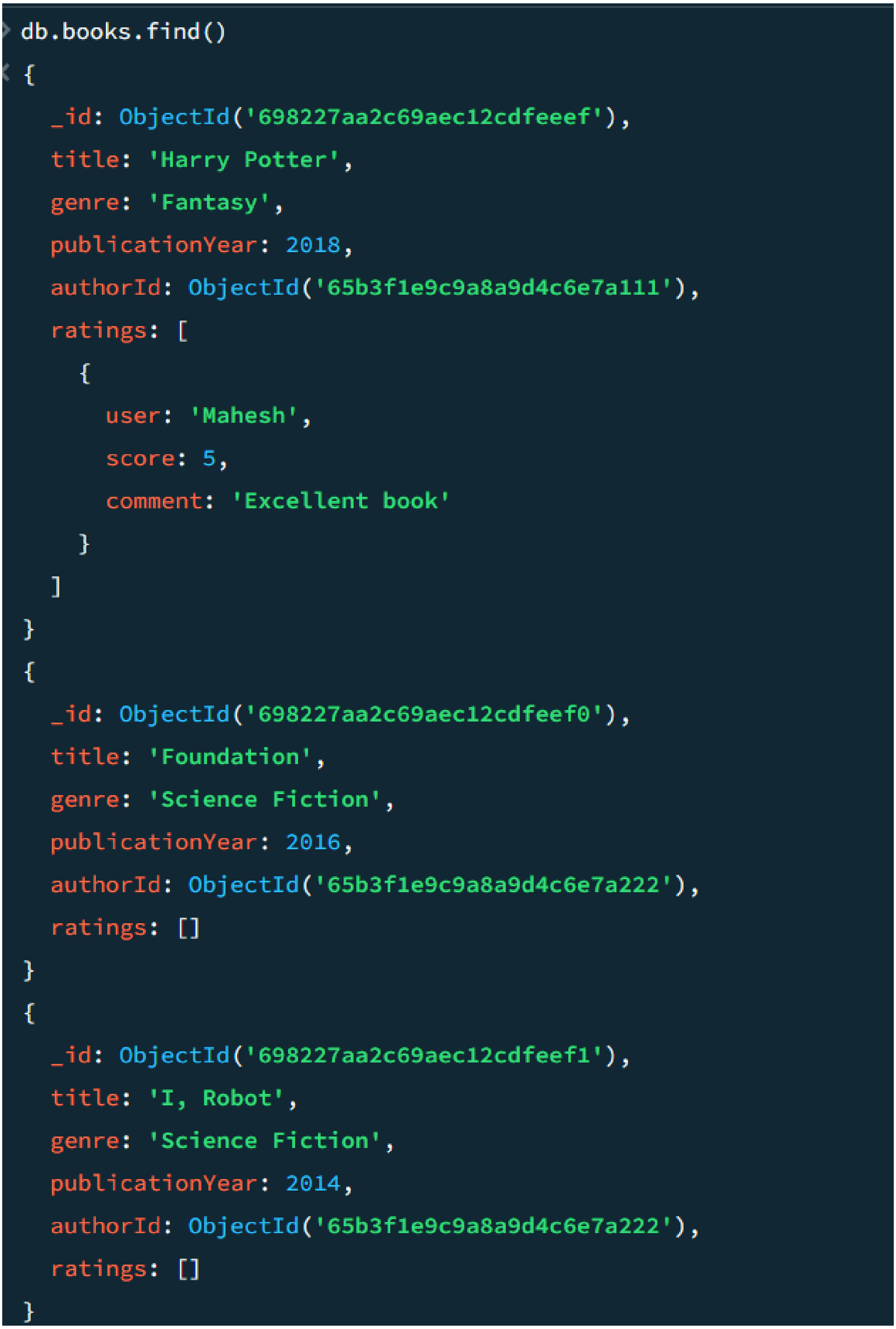
db.createCollection("authors")

db.createCollection("books")

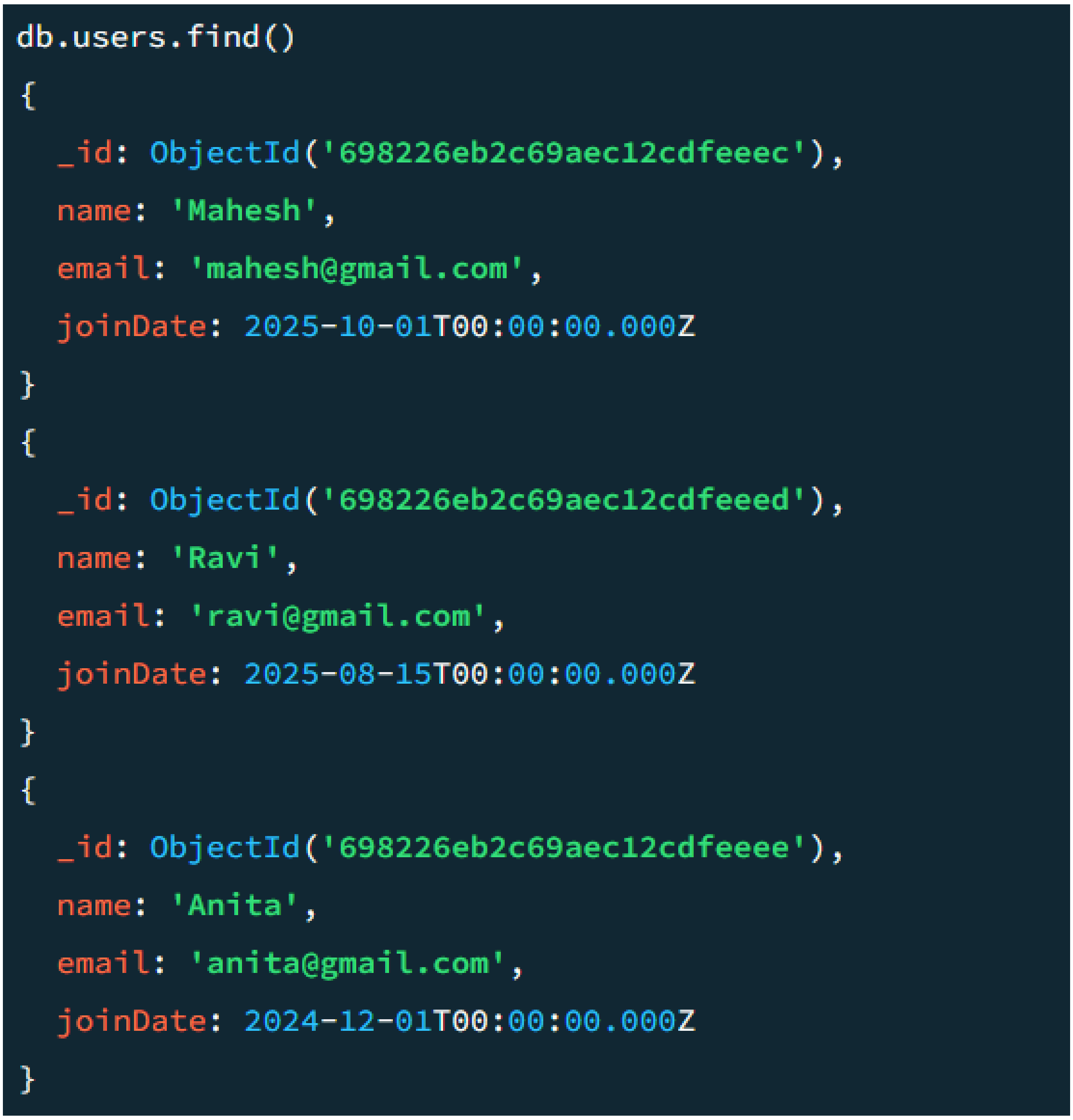
db.createCollection("users")



```
db.authors.find()
{
  _id: ObjectId('698226202c69aec12cdfeee9'),
  name: 'J. K. Rowling',
  nationality: 'British',
  birthYear: 1965
}
{
  _id: ObjectId('698226202c69aec12cdfeeea'),
  name: 'Isaac Asimov',
  nationality: 'American',
  birthYear: 1920
}
{
  _id: ObjectId('698226202c69aec12cdfeeeb'),
  name: 'George R. R. Martin',
  nationality: 'American',
  birthYear: 1948
}
```



```
db.books.find()
{
  _id: ObjectId('698227aa2c69aec12cdfeeef'),
  title: 'Harry Potter',
  genre: 'Fantasy',
  publicationYear: 2018,
  authorId: ObjectId('65b3f1e9c9a8a9d4c6e7a111'),
  ratings: [
    {
      user: 'Mahesh',
      score: 5,
      comment: 'Excellent book'
    }
  ]
}
{
  _id: ObjectId('698227aa2c69aec12cdfeef0'),
  title: 'Foundation',
  genre: 'Science Fiction',
  publicationYear: 2016,
  authorId: ObjectId('65b3f1e9c9a8a9d4c6e7a222'),
  ratings: []
}
{
  _id: ObjectId('698227aa2c69aec12cdfeef1'),
  title: 'I, Robot',
  genre: 'Science Fiction',
  publicationYear: 2014,
  authorId: ObjectId('65b3f1e9c9a8a9d4c6e7a222'),
  ratings: []
}
```



```
db.users.find()
{
  _id: ObjectId('698226eb2c69aec12cdfeeec'),
  name: 'Mahesh',
  email: 'mahesh@gmail.com',
  joinDate: 2025-10-01T00:00:00.000Z
}
{
  _id: ObjectId('698226eb2c69aec12cdfeeed'),
  name: 'Ravi',
  email: 'ravi@gmail.com',
  joinDate: 2025-08-15T00:00:00.000Z
}
{
  _id: ObjectId('698226eb2c69aec12cdfeeee'),
  name: 'Anita',
  email: 'anita@gmail.com',
  joinDate: 2024-12-01T00:00:00.000Z
}
```

User Story 2 — CRUD Operations

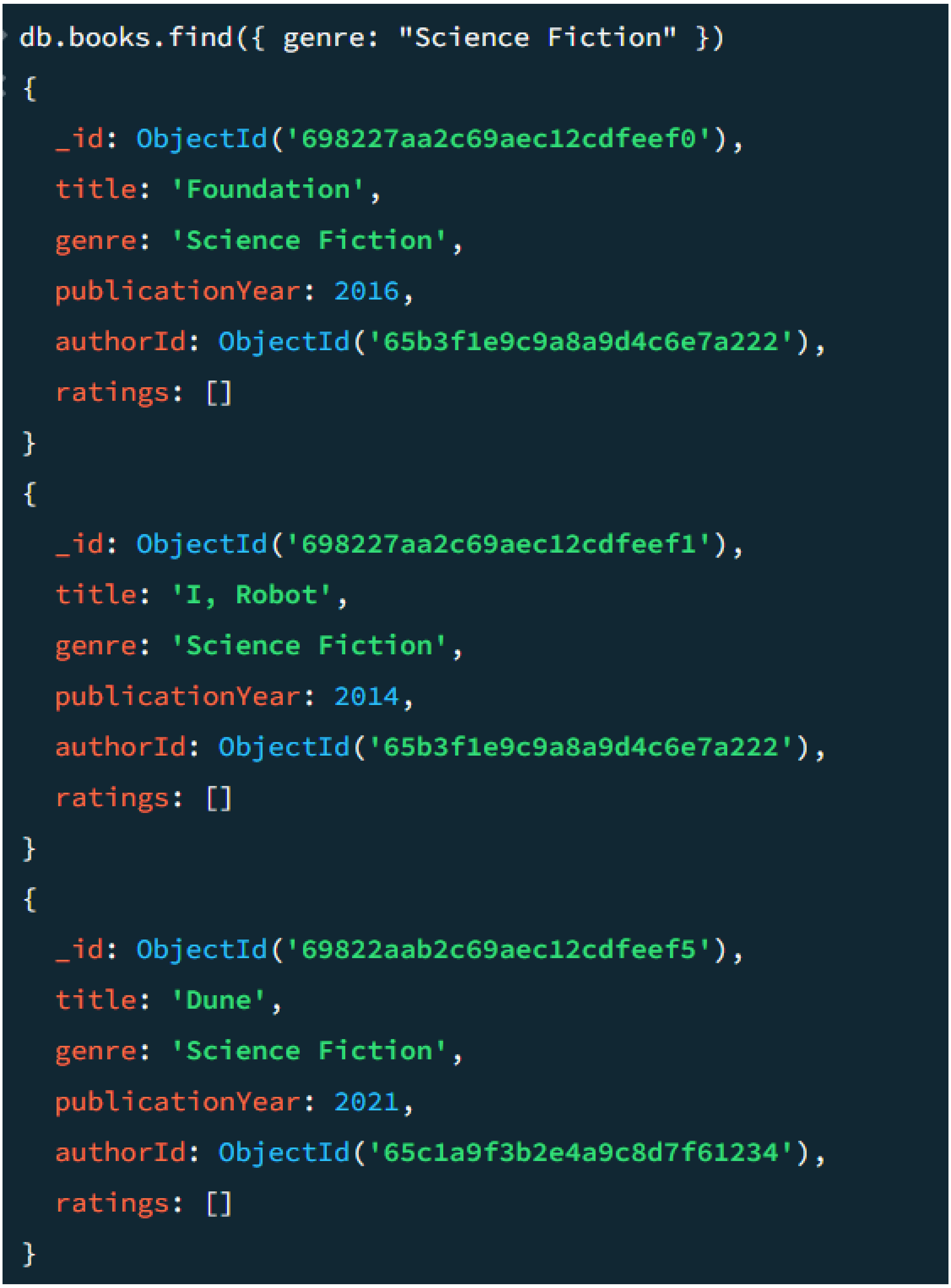
1. Insert new users and books into the database.

```
db.users.insertOne({
  name: "Suresh",
  email: "suresh@gmail.com",
  joinDate: new Date()
})
```

```
db.books.insertOne({
  title: "Dune",
  genre: "Science Fiction",
  publicationYear: 2021,
  authorId: ObjectId("65c1a9f3b2e4a9c8d7f61234"),
  ratings: []
})
```

2. Retrieve all books of the genre “Science Fiction”.

```
db.books.find({ genre: "Science Fiction" })
```



```
db.books.find({ genre: "Science Fiction" })
{
  _id: ObjectId('698227aa2c69aec12cdfeef0'),
  title: 'Foundation',
  genre: 'Science Fiction',
  publicationYear: 2016,
  authorId: ObjectId('65b3f1e9c9a8a9d4c6e7a222'),
  ratings: []
}
{
  _id: ObjectId('698227aa2c69aec12cdfeef1'),
  title: 'I, Robot',
  genre: 'Science Fiction',
  publicationYear: 2014,
  authorId: ObjectId('65b3f1e9c9a8a9d4c6e7a222'),
  ratings: []
}
{
  _id: ObjectId('69822aab2c69aec12cdfeef5'),
  title: 'Dune',
  genre: 'Science Fiction',
  publicationYear: 2021,
  authorId: ObjectId('65c1a9f3b2e4a9c8d7f61234'),
  ratings: []
}
```

3. Update the publicationYear of one book.

```
db.books.updateOne(
  { title: "Foundation" },
  { $set: { publicationYear: 2020 } })
```

4. Delete one user record from the collection.

```
db.users.deleteOne({ name: "Ravi" })
```

5. Add a new rating to a book document using the $push operator.

```
db.books.updateOne(
  { title: "Game of Thrones" },
  {
    $push: {
      ratings: {
        user: "Anita",
        score: 5,
        comment: "Great fantasy novel"
      }
    }
  }
)
```

User Story 3 — Querying & Filtering

1. Retrieve all books published after 2015.

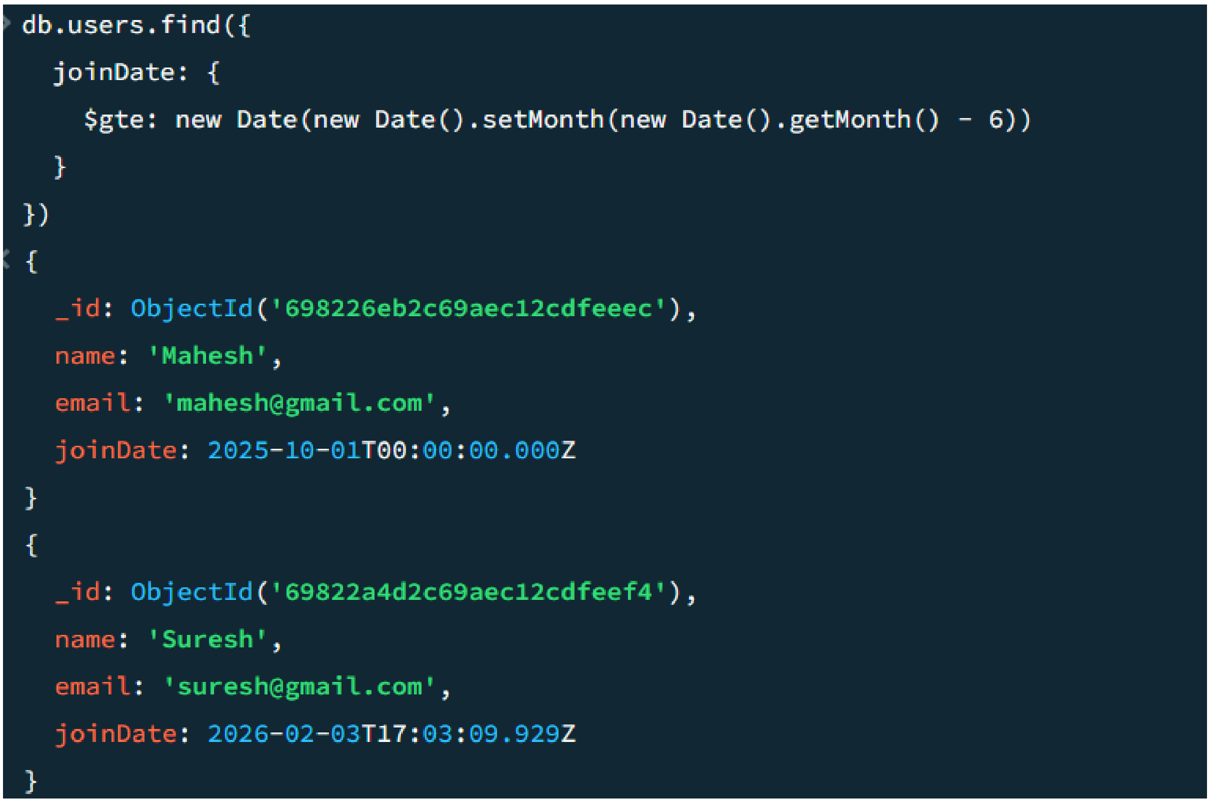
```
db.books.find({ publicationYear: { $gt: 2015 } })
```

2. Find authors who have written books in the “Fantasy” genre.

```
db.books.distinct("authorId", { genre: "Fantasy" })
```

3. Retrieve all users who joined within the last 6 months.

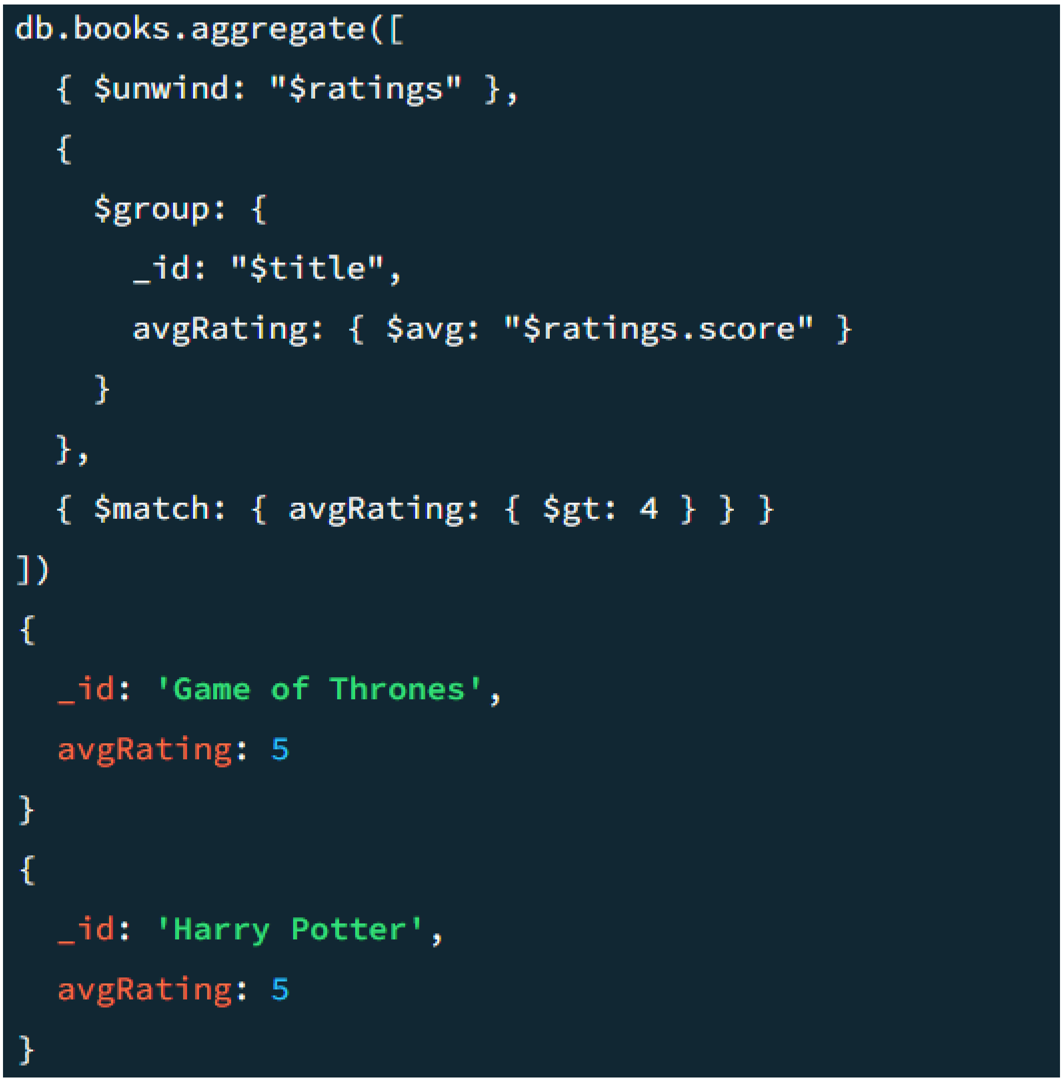
```
db.users.find({
  joinDate: {
    $gte: new Date(new Date().setMonth(new
Date().getMonth() - 6))
  }
})
```



```
db.users.find({
  joinDate: {
    $gte: new Date(new Date().setMonth(new Date().getMonth() - 6))
  }
})
{
  _id: ObjectId('698226eb2c69aec12cdfeeec'),
  name: 'Mahesh',
  email: 'mahesh@gmail.com',
  joinDate: 2025-10-01T00:00:00.000Z
}
{
  _id: ObjectId('69822a4d2c69aec12cdfeef4'),
  name: 'Suresh',
  email: 'suresh@gmail.com',
  joinDate: 2026-02-03T17:03:09.929Z
}
```

4. Find books with an average rating greater than 4.

```
db.books.aggregate([
  { $unwind: "$ratings" },
  {
    $group: {
      _id: "$title",
      avgRating: { $avg: "$ratings.score" }
    }
  },
  { $match: { avgRating: { $gt: 4 } } }])
```



```
db.books.aggregate([
  { $unwind: "$ratings" },
  {
    $group: {
      _id: "$title",
      avgRating: { $avg: "$ratings.score" }
    }
  },
  { $match: { avgRating: { $gt: 4 } } }
])
{
  _id: 'Game of Thrones',
  avgRating: 5
}
{
  _id: 'Harry Potter',
  avgRating: 5
}
```