Day 8 coding assessment

Name: Arpit Mahalle

Output of User Story 1:

Authors:

Example Query	Purpose	Field(s) Involved
db.authors.find({ name: "J.K. Rowling" })	Lookup by name	name
db.authors.find({ nationality: "British" })	Filter by nationality	nationality
db.authors.find({ birthYear: { $lt: 1950 } })	Older authors (less frequent)	birthYear

Frequently queried fields from “Authors” collection.: **name, nationality**

Books:

Example Query	Purpose	Field(s) Involved
db.books.find({ genre: "Fantasy" })	Filter books by genre	genre
db.books.find({ authorId: 1 })	Find all books by an author	authorId
db.books.find({ "ratings.score": { $gte: 4.5 } })	Get highly-rated books	ratings.score
db.books.find({ publicationYear: 1998 })	Filter by year (less frequent)	publicationYear

Frequently queried fields from “Books” collection**: genre, authorId, ratings.score**

Users:

Example Query	Purpose	Field(s) Involved
db.users.find({ email: "shyam@example.com" })	Login or lookup by email	email
db.users.find({ joinDate: { $gte: ISODate("2023-01-01") } })	Filter by join date (less frequent)	joinDate

Frequently queried fields from “Users” collection: **email.**

```
> db.books.createIndex({ genre: 1 });
< genre_1
> db.books.createIndex({ authorId: 1 });
< authorId_1
> db.books.createIndex({ "ratings.score": 1 });
< ratings.score_1
> db.users.createIndex({ email: 1 });
< email_1
```

Index is created on these fields.

Queries:

db.books.dropIndex({ "ratings.score": 1 })

db.books.getIndexes()

```
< [
    { v: 2, key: { _id: 1 }, name: '_id_' },
    { v: 2, key: { genre: 1 }, name: 'genre_1' },
    { v: 2, key: { authorId: 1 }, name: 'authorId_1' }
  ]
```

ratings.score Index is dropped and the remaining fields are displayed.

User story 2 output:

```
db.books.aggregate([
  { $unwind: "$ratings" }, // Flatten ratings array
  {
    $group: {
      _id: "$title",
      averageRating: { $avg: "$ratings.score" }
    }
  },
  {
    $project: {
      _id: 0,
      title: "$_id",
      averageRating: { $round: ["$averageRating", 2] }
    }
  }
]);
{
  title: "Harry Potter and the Sorcerer's Stone",
  averageRating: 4.9
}
{
  title: 'Harry Potter and the Chamber of Secrets',
  averageRating: 4.8
}
{
  title: 'Kafka on the Shore',
  averageRating: 4.5
}
{
  title: 'A Game of Thrones',
  averageRating: 4.7
}
```

Calculating the average rating per book using $unwind, $group, and $avg.

```
db.books.aggregate([
  { $unwind: "$ratings" },
  {
    $group: {
      _id: "$title",
      averageRating: { $avg: "$ratings.score" }
    }
  },
  { $sort: { averageRating: -1 } },  // Sort in descending order
  { $limit: 3 },                     // Pick top 3
  {
    $project: {
      _id: 0,
      title: "$_id",
      averageRating: 1
    }
  }
]);
{
  averageRating: 4.9,
  title: "Harry Potter and the Sorcerer's Stone"
}
{
  averageRating: 4.8,
  title: 'Harry Potter and the Chamber of Secrets'
}
{
  averageRating: 4.7,
  title: 'A Game of Thrones'
}
```

Retrieving the top 3 highest-rated books.

```
db.books.aggregate([
  {
    $group: {
      _id: "$genre",
      totalBooks: { $sum: 1 }
    }
  },
  {
    $project: {
      _id: 0,
      genre: "$_id",
      totalBooks: 1
    }
  }
]);
{
  totalBooks: 4,
  genre: 'Fantasy'
}
{
  totalBooks: 1,
  genre: 'Magical Realism'
}
```

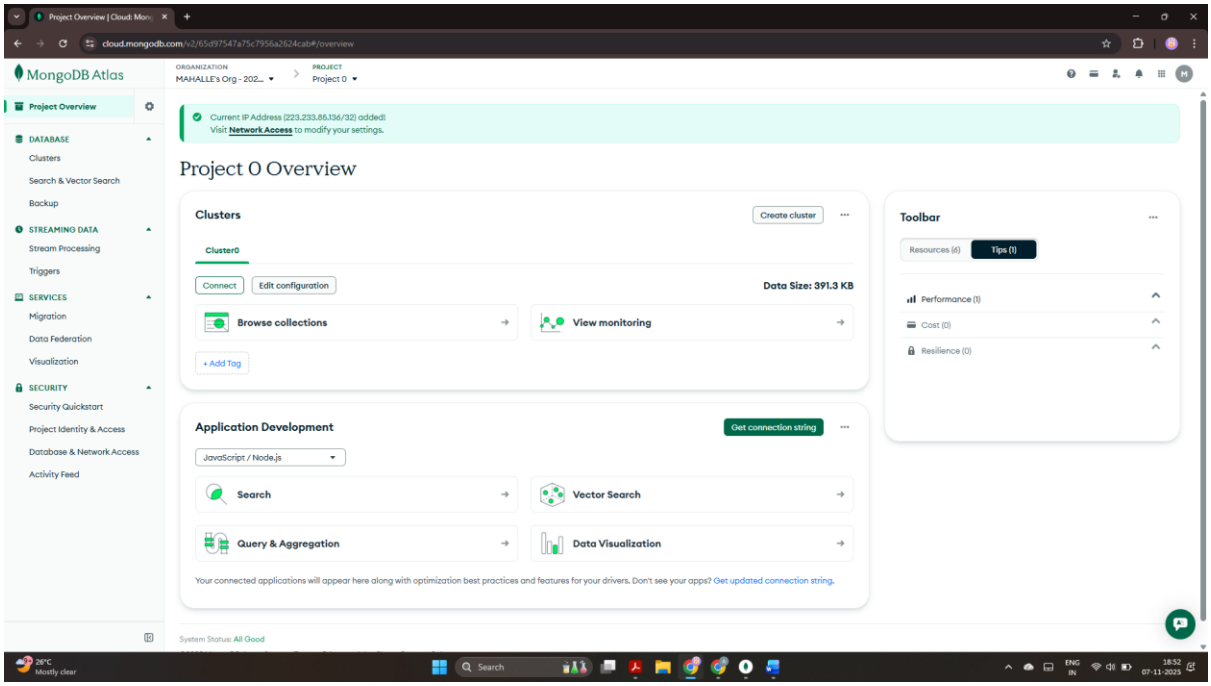
Counting the number of books published per genre.

Find authors who have more than 2 books published: The output is empty because no author has more than2 books published.

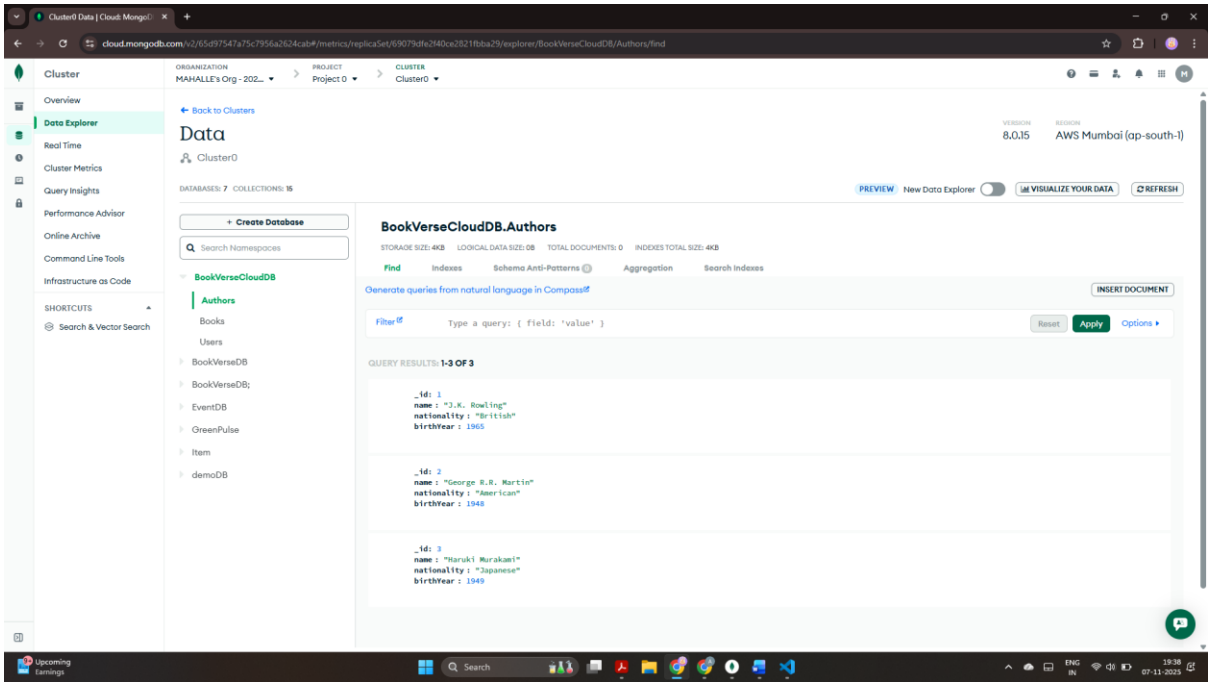
```
db.books.aggregate([
  { $unwind: "$ratings" },
  {
    $group: {
      _id: "$authorId",
      totalRewardPoints: { $sum: "$ratings.score" }
    }
  },
  {
    $lookup: {
      from: "authors",
      localField: "_id",
      foreignField: "_id",
      as: "authorDetails"
    }
  },
  { $unwind: "$authorDetails" },
  {
    $project: {
      _id: 0,
      authorName: "$authorDetails.name",
      totalRewardPoints: { $round: ["$totalRewardPoints", 2] }
    }
  }
]);
{
  authorName: 'Haruki Murakami',
  totalRewardPoints: 4.5
}
{
  authorName: 'J.K. Rowling',
  totalRewardPoints: 9.7
}
{
  authorName: 'George R.R. Martin',
  totalRewardPoints: 4.7
}
```

Displaying the total reward points (sum of all ratings) received by each author.

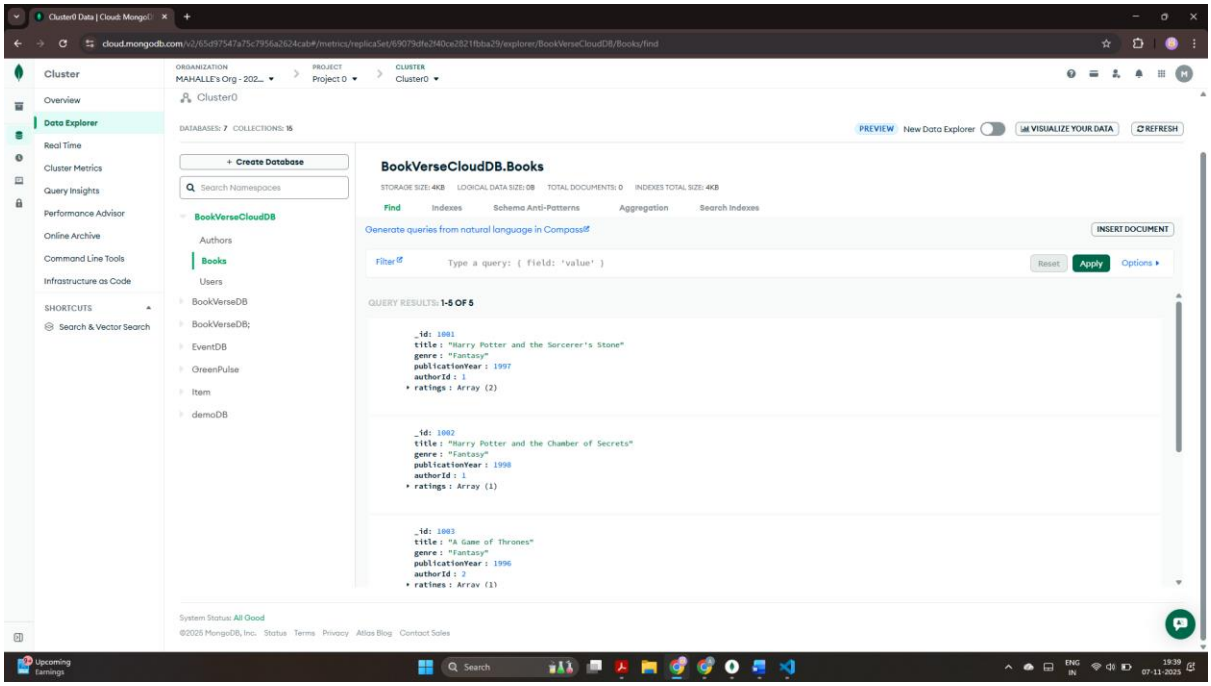
User Story 3 output:



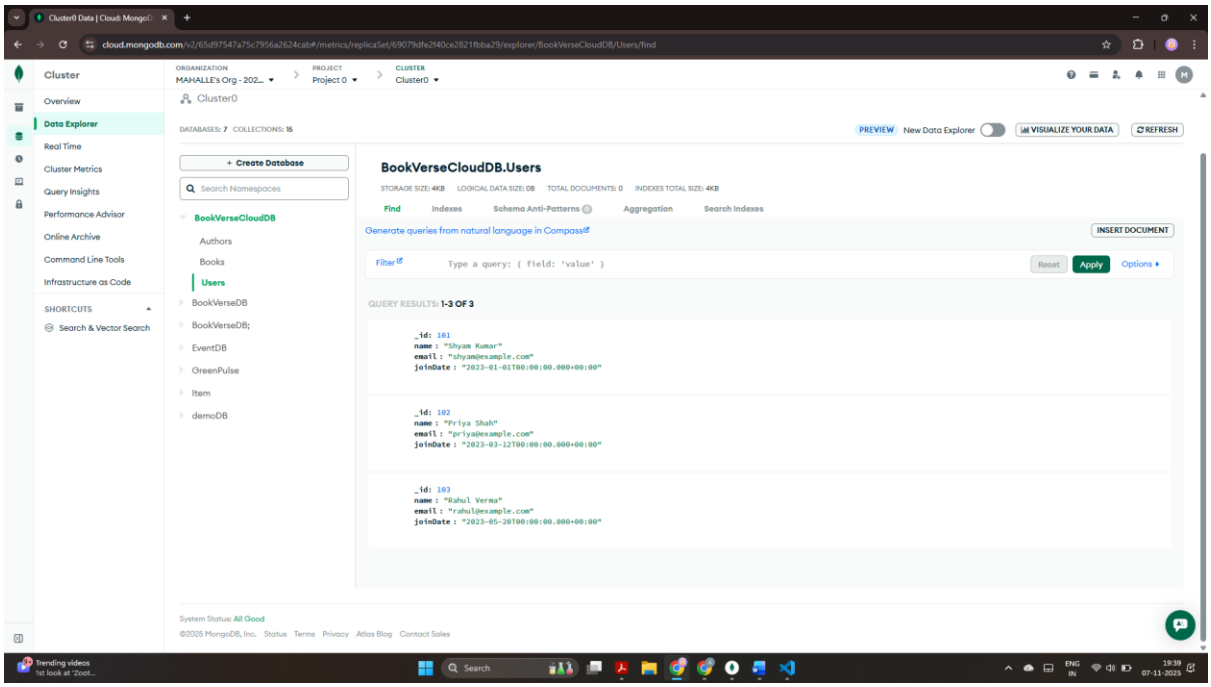
Free cluster is created. The name of the cluster is “Cluster0”.



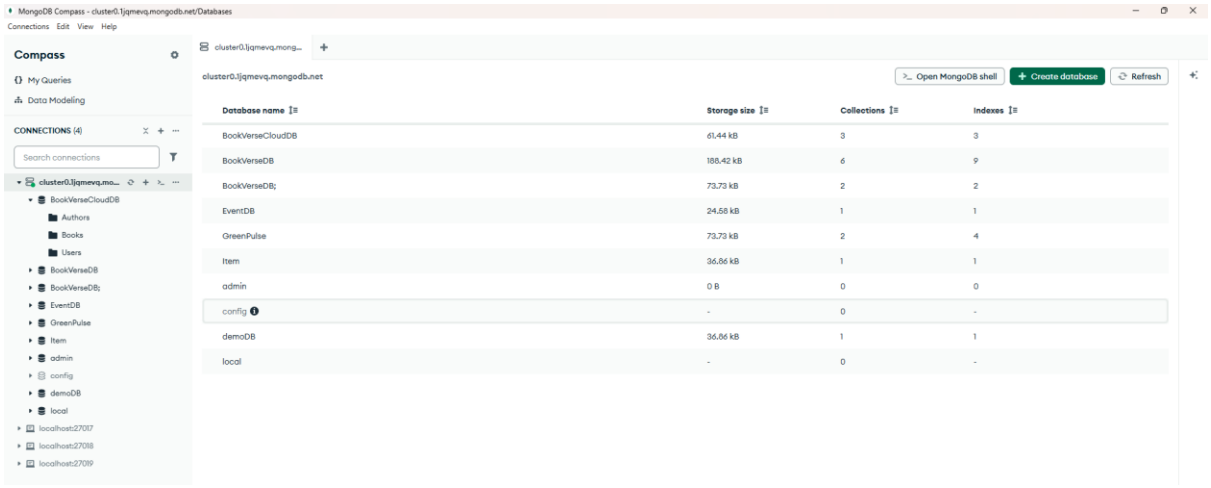
Data in “Authors” collection is inserted.



Data in “Books” collection is inserted.



Data in “Users” collection is inserted.



Connected my cluster using the connection string in MongoDB Compass.