**MongoDB Practice Assignment**

**Assignment Brief**

**Background:**  
You are hired to build the backend database structure for a small online book review app. Users can register, write reviews on books, and rate them. Your job is to build the MongoDB collections and perform various operations to demonstrate your skills.

**Part 1: Setup & Fake Data Creation**

1. **Create 2 collections:**
   * **users** (user info)
   * **reviews** (book reviews)
2. **Using the Faker library, generate and insert:**
   * **50 users** into the users collection. Each user should have:
     + name (fake full name)
     + email (fake unique email)
     + age (random number between 18 and 70)
     + isPremiumMember (boolean)
   * **200 reviews** into the reviews collection. Each review should have:
     + userId (pick randomly from existing user IDs)
     + bookTitle (random title)
     + reviewText (random paragraph)
     + rating (random integer from 1 to 5)
     + createdAt (random past date)

**Part 2: CRUD Operations Practice**

**Read Operations (Queries)**

* Find all users older than 30.
* Find premium members who are below 25 years old.
* Get all reviews with a rating greater than 3.
* Find reviews where the book title starts with **'The'**.
* Retrieve reviews made in the last 30 days.

**Update Operations**

* Mark all users older than 65 as isPremiumMember: true.
* Increase the rating of reviews made before 2023 by 1 point (max 5).
* Change the email domain of all users from @oldmail.com to @newmail.com.

**Delete Operations**

* Delete all users younger than 20 and not premium members.
* Delete all reviews with rating less than or equal to 2.

**Part 3: Advanced Query Practice**

**Logical Operators**

* Find users who are **either** premium members **or** above 50 years old.
* Find reviews where the rating is **either 5 or 1**.

**Combination Queries**

* Get users who are premium members **AND** between ages 30 to 50.
* Get reviews for books starting with **'A'** or **'B'** with ratings above 3.

**Bonus (Optional but Recommended for Mastery)**

**Aggregation Practice**

* Group reviews by bookTitle and count how many reviews each book has.
* Find the **average rating** per book.
* Get the **top 5 books** with the highest average rating.

**Submission**

* Submit your code (Python or Node.js scripts).
* Submit exported results of your queries (copy-paste JSON output or screenshots).
* **Bonus:** Create a short doc explaining what you learned or found challenging.