📘 Homework: MongoDB Queries & Operations

Setup

Create a database called school and a collection students. Insert at least 8 sample students with fields like:

```
{
  name: "Ali",
  age: 17,
  grade: 11,
  gpa: 3.5,
  subjects: ["Math", "English", "Science"],
  address: { city: "Jerusalem", street: "King George" }
}
```

Part 1: Basic Queries

1. Insert 5 new students into the students collection.
2. Retrieve all students whose age is greater than 16.
3. Find students who study "Math" as one of their subjects.
4. Show only the name and gpa fields of all students (exclude _id).
5. Update a student's GPA to 4.0 where name is "Ali".
6. Delete one student whose grade is 9.

Part 2: Aggregation & Advanced Queries

7. Count how many students are in each grade.
8. Find the average gpa of students per grade.
9. Retrieve the top 3 students with the highest GPA.
10. Find all students who live in "Jerusalem".
11. Sort students by age in descending order.
12. Find students who are not taking "Science".
13. Update all students in grade 10 to increase their GPA by 0.2.
14. Remove the "address" field from all documents.

Part 3: More Challenging

15. Create a collection teachers. Insert some teachers with name, subject, and yearsExperience.
16. Perform a $lookup between students and teachers so each student shows which teacher teaches one of their subjects.
17. Create an index on the gpa field. Run a query using that index.
18. Create a transaction that:

- Inserts a new student
- Updates another student's GPA
- Deletes a third student

19. Disable auto-commit (session.startTransaction()) and test a rollback.
20. Write a function getTopStudent() that returns the student with the highest GPA.
21. Demonstrate cascade-like behavior: Create two collections classes and students. When a class is deleted, delete all students in that class (using transactions or manual $in query).
22. Use $setUnion to merge two arrays of subjects for a student.