**🧱 Basic CRUD Operations**

1. **Create a new database named mydatabase.**
2. **Create a collection named students.**
3. **Insert two documents into the students collection with fields such as name, age, and subject.**
4. **Retrieve all documents from the students collection.**
5. **Retrieve only the names of students from the students collection.**
6. **Find all students who are 25 years old.**
7. **Update the age of a specific student in the students collection.**
8. **Add a new field, grade, with the value "A" to all documents in the students collection.**
9. **Delete a specific student from the students collection.**
10. **Remove the grade field from all documents in the students collection.**[aryatechno.com](https://www.aryatechno.com/topics/mongodb-exercises?utm_source=chatgpt.com)

**🔄 Update and Delete Operations**

1. **Update the subject of a student named "John" to "Physics".**
2. **Increment the age of all students by 1.**
3. **Rename the subject field to course in all documents.**
4. **Delete all students who are older than 30.**
5. **Remove all documents where the grade field is missing.**

**🔍 Querying and Filtering**

1. **Find all students who are enrolled in "Math" or "History".**
2. **Retrieve students whose names start with the letter "A".**
3. **Find students whose age is between 20 and 25.**
4. **Retrieve students who do not have a grade field.**
5. **Find students who are either 22 or 24 years old.**

**📊 Aggregation Framework**

1. **Calculate the average age of students.**
2. **Group students by subject and count the number of students in each subject.**
3. **Find the student with the highest age.**
4. **Calculate the total number of students.**
5. **Find the student with the minimum age.**[aryatechno.com+1github.com+1](https://www.aryatechno.com/topics/mongodb-exercises?utm_source=chatgpt.com)

**🗂️ Indexing and Performance**

1. **Create an index on the name field in the students collection.**
2. **Create a compound index on age and subject.**
3. **List all indexes on the students collection.**
4. **Drop the index on the name field.**
5. **Explain the execution plan of a query to see if an index is being used.**[aryatechno.com+1aryatechno.com+1](https://www.aryatechno.com/topics/mongodb-exercises?utm_source=chatgpt.com)

**🔠 Text Search and Regular Expressions**

1. **Create a text index on the name and subject fields.**
2. **Perform a text search for students with the keyword "Math".**
3. **Find students whose name contains the substring "Ali".**
4. **Find students whose subject ends with "History".**
5. **Find students whose name starts with "J".**[aryatechno.com](https://www.aryatechno.com/topics/mongodb-exercises?utm_source=chatgpt.com)

**📍 Geospatial Queries**

1. **Create a collection named locations with latitude and longitude fields.**
2. **Insert documents representing locations with latitude and longitude fields.**
3. **Find locations near a specific point using geospatial queries.**
4. **Create a 2dsphere index on the location field.**
5. **Find locations within a certain distance from a point.**[aryatechno.com+4aryatechno.com+4aryatechno.com+4](https://www.aryatechno.com/topics/mongodb-exercises?utm_source=chatgpt.com)

**🧪 Advanced Aggregation**

1. **Use the $match stage to filter documents.**
2. **Use the $group stage to group documents by a field.**
3. **Use the $sort stage to sort documents.**
4. **Use the $project stage to include or exclude fields.**
5. **Use the $lookup stage to perform a left outer join.**

**🛠️ Data Validation and Schema Design**

1. **Define a schema validation rule for the students collection to ensure age is an integer.**
2. **Create a collection with embedded documents for address and phone numbers.**
3. **Design a schema for an e-commerce platform with collections for products, customers, and orders.**
4. **Implement a one-to-many relationship between authors and books using references.**
5. **Design a schema for a blogging platform with collections for posts, comments, and tags.**