MongoDB Query Implementation Questions

Databases and Collections

1. Create a new database named `bookstore`.
2. Create a new collection named `authors` in the `bookstore` database.
3. Create collections for `books`, `customers`, `orders`, and `orderDetails` in the `bookstore` database.
4. List all databases and collections in MongoDB.

Documents

5. Insert a new author document into the `authors` collection.
6. Retrieve all documents from the `authors` collection.
7. Update an author's document in the `authors` collection.
8. Delete an author's document from the `authors` collection.

CRUD Operations (Create, Read, Update, Delete)

9. Insert a new book document into the `books` collection.
10. Retrieve all books from the `books` collection.
11. Update the price of a book in the `books` collection.
12. Delete a book from the `books` collection.
13. Insert a new customer document into the `customers` collection.
14. Retrieve all customers from the `customers` collection.
15. Update a customer's email in the `customers` collection.
16. Delete a customer from the `customers` collection.
17. Insert a new order document into the `orders` collection.
18. Retrieve all orders from the `orders` collection.
19. Insert a new order detail document into the `orderDetails` collection.
20. Retrieve all order details from the `orderDetails` collection.

Array Operations

21. Insert a document with an array field into the `books` collection.
22. Retrieve documents from the `books` collection where an array field contains a specific value.
23. Add an element to an array field in a document of the `books` collection.
24. Remove an element from an array field in a document of the `books` collection.

Operators

25. Retrieve books where the price is greater than 20.

26. Retrieve books where the genre is either 'Science Fiction' or 'Fantasy'.
27. Retrieve authors where the `lastName` starts with 'A'.
28. Retrieve orders placed between two dates.

MongoDB Cursor

29. Iterate over a cursor to retrieve all books in the `books` collection.

Projection

30. Retrieve only the `title` and `price` fields from the `books` collection.
31. Retrieve all fields except `price` from the `books` collection.

Lookup

32. Perform a lookup to join the `books` collection with the `authors` collection.
33. Perform a lookup to join the `orders` collection with the `customers` collection.

Group

34. Group books by genre and retrieve the count of books in each genre.
35. Group orders by `customerId` and retrieve the total quantity of books ordered by each customer.

Transaction

36. Demonstrate how to perform a multi-document transaction in MongoDB.

Regex

37. Retrieve authors where the `lastName` matches a specific regex pattern.
38. Retrieve books where the `title` matches a specific regex pattern.

Querying Documents

39. Retrieve books where the `title` contains the word 'MongoDB'.
40. Retrieve customers who joined after a specific date.
41. Retrieve orders placed by a specific customer.

Indexing

42. Create an index on the `lastName` field in the `authors` collection.
43. Create a compound index on the `genre` and `price` fields in the `books` collection.
44. List all indexes on the `books` collection.

Aggregation Framework

45. Use the aggregation framework to calculate the average price of books in each genre.

46. Use the aggregation framework to calculate the total sales for each book.
47. Use the aggregation framework to find the top 3 most expensive books.

Embedded and Referenced Documents

48. Retrieve books with embedded author details from the `books` collection.
49. Retrieve books with referenced author details using a lookup.

Security Features

50. Demonstrate how to create a user with read-only access to the `bookstore` database.
51. Demonstrate how to create a user with read and write access to the `bookstore` database.
52. Explain the authentication mechanisms supported by MongoDB.

Advanced Topics

53. Demonstrate the use of MongoDB's Change Streams for real-time notifications.
54. Use MongoDB's GridFS to store and retrieve large files.

Optimization and Performance

55. Demonstrate the use of explain() to analyze query performance.

Backup and Restoration

56. Demonstrate how to restore a MongoDB database from a backup.

Geographic Queries

57. Demonstrate how to perform a geospatial query to find locations near a specific point.

Data Modeling

58. Design a schema for a blog application using MongoDB.

Full-Text Search

59. Implement a full-text search in MongoDB for blog posts.

Change Management

60. Demonstrate schema migrations in MongoDB.

Aggregation Pipeline Optimization

61. Optimize an existing aggregation pipeline in MongoDB for better performance.

GridFS

62. Demonstrate how to store and retrieve large files using GridFS.

Error Handling and Troubleshooting

63. Explain how to monitor MongoDB performance and diagnose issues.

Compatibility and Integration

64. Demonstrate integration of MongoDB with a Node.js application.

Deployment Strategies

65. Demonstrate how to deploy a MongoDB replica set.

Backup Strategies

66. Demonstrate how to automate MongoDB backups using MongoDB Atlas.

Scaling Strategies

67. Demonstrate how to add shards to an existing MongoDB cluster.

Data Encryption

68. Demonstrate how to enable encryption in MongoDB.

Monitoring and Alerting

69. Demonstrate how to use MongoDB Ops Manager for monitoring.

Deployment in Cloud Environments

70. Discuss strategies for deploying MongoDB in cloud environments.

This revised list focuses purely on practical MongoDB implementation questions across a wide range of topics, ensuring comprehensive coverage for learners.