MongoDB Interview — 50 High■Yield Q&A; (with quick answers & examples)

This document provides 50 MongoDB interview-style questions with concise answers and command/code examples.

Q1. What is MongoDB?

Quick answer: A NoSQL, document-oriented database storing data in BSON (binary JSON).

Example: db.version()

Q2. Difference between MongoDB and MySQL?

Quick answer: MongoDB: NoSQL, schema-less, JSON-like docs. MySQL: relational, structured schema.

Example: db.users.find({name:'John'})

Q3. What is a collection?

Quick answer: Group of MongoDB documents, equivalent to table in SQL.

Example: db.createCollection('users')

Q4. What is a document?

Quick answer: A record in MongoDB stored as BSON (JSON-like).

Example: { name:'Alice', age:25 }

Q5. What is ObjectId in MongoDB?

Quick answer: Unique 12-byte identifier for each document.

Example: ObjectId('507f1f77bcf86cd799439011')

Q6. How to insert documents?

Quick answer: Use insertOne or insertMany.

Example: db.users.insertOne({name:'Bob'})

Q7. How to query documents?

Quick answer: Use find with query filters.

Example: db.users.find({age:{$gt:20}})

Q8. Difference between findOne and find?

Quick answer: findOne returns single doc; find returns cursor of docs.

Example: db.users.findOne({name:'Sam'})

Q9. What are operators in MongoDB?

Quick answer: $gt, $lt, $in, $and, $or etc.

Example: db.users.find({age:{$in:[20,25]}})

Q10. What is updateOne vs updateMany?

Quick answer: updateOne updates first match, updateMany updates all.

Example: db.users.updateOne({name:'Bob'},{$set:{age:30}})

Q11. What is deleteOne vs deleteMany?

Quick answer: deleteOne removes first match; deleteMany removes all.

Example: db.users.deleteMany({inactive:true})

Q12. What is aggregation in MongoDB?

Quick answer: Framework to process data and return computed results.

Example: db.sales.aggregate([{$group:{_id:'$item', total:{$sum:'$price'}}}])

Q13. Difference between $match and $group?

Quick answer: $match filters docs. $group groups docs for aggregation.

Example: db.orders.aggregate([{$match:{status:'A'}},{$group:{_id:'$cust_id',total:{$sum:'$amount'}}}])

Q14. What is $project in aggregation?

Quick answer: Specifies fields to include/exclude or compute new fields.

Example: db.users.aggregate([{$project:{name:1,year:{$year:'$dob'}}}])

Q15. What are indexes in MongoDB?

Quick answer: Structures that improve query performance.

Example: db.users.createIndex({name:1})

Q16. Difference between single-field and compound index?

Quick answer: Single: one field. Compound: multiple fields.

Example: db.users.createIndex({last:1,first:1})

Q17. What is a text index?

Quick answer: Index for text search.

Example: db.articles.createIndex({content:'text'})

Q18. How to perform text search?

Quick answer: Use $text operator.

Example: db.articles.find({$text:{$search:'coffee'}})

Q19. What are capped collections?

Quick answer: Fixed-size collections that overwrite oldest data when full.

Example: db.createCollection('logs',{capped:true,size:10000})

Q20. What is sharding?

Quick answer: Horizontal scaling by distributing data across shards.

Example: sh.enableSharding('mydb')

Q21. What is replication?

Quick answer: Copies of data on multiple servers for redundancy and HA.

Example: rs.initiate()

Q22. Difference between replica set and sharding?

Quick answer: Replica set: copies for redundancy. Sharding: partitions for scaling.

Example: rs.add('node2:27017')

Q23. What is primary and secondary in replica set?

Quick answer: Primary accepts writes. Secondary replicates data.

Example: rs.status()

Q24. What is journaling in MongoDB?

Quick answer: Durability feature to recover data after crash.

Example: enabled by default in WiredTiger.

Q25. What is WiredTiger?

Quick answer: Default storage engine in MongoDB with compression and concurrency control.

Example: db.serverStatus().storageEngine

Q26. Difference between MongoDB and Cassandra?

Quick answer: Mongo: document store. Cassandra: wide-column store.

Example: Both NoSQL but different data models.

Q27. What is schema validation?

Quick answer: Enforce rules on documents in collection.

Example: db.createCollection('users',{validator:{ $jsonSchema:{required:['name']}}})

Q28. How to do transactions in MongoDB?

Quick answer: Multi-document transactions supported in replica sets/sharded clusters.

Example: session.startTransaction();

Q29. What are change streams?

Quick answer: Real-time feed of changes to collections/databases.

Example: db.collection.watch()

Q30. What is difference between Mongo shell and Compass?

Quick answer: Shell: CLI. Compass: GUI for MongoDB.

Example: Compass provides visual exploration.

Q31. What is GridFS?

Quick answer: Storage spec for large files by splitting into chunks.

Example: db.fs.files, db.fs.chunks

Q32. How to ensure uniqueness of a field?

Quick answer: Use unique index.

Example: db.users.createIndex({email:1},{unique:true})

Q33. What is difference between $lookup and JOIN?

Quick answer: $lookup simulates join between collections.

Example: db.orders.aggregate([{$lookup:{from:'users',localField:'uid',foreignField:'_id',as:'user'}}])

Q34. What is difference between find() and aggregate()?

Quick answer: find for simple queries. aggregate for advanced pipelines.

Example: db.users.find({age:{$gt:20}}) vs db.users.aggregate([{$match:{age:{$gt:20}}}])

Q35. What is explain() in MongoDB?

Quick answer: Shows query execution plan.

Example: db.users.find({name:'Bob'}).explain()

Q36. What is MongoDB Atlas?

Quick answer: Managed cloud database service for MongoDB.

Example: Atlas provides backups, monitoring, scaling.

Q37. What is difference between save() and insert()?

Quick answer: insert inserts new docs. save inserts or updates existing by _id.

Example: db.users.save({_id:1,name:'Joe'})

Q38. What are TTL indexes?

Quick answer: Indexes that auto-delete documents after time.

Example: db.sessions.createIndex({createdAt:1},{expireAfterSeconds:3600})

Q39. What are sparse indexes?

Quick answer: Indexes only documents with the field present.

Example: db.users.createIndex({nickname:1},{sparse:true})

Q40. What are covered queries?

Quick answer: Queries answered using only index without scanning docs.

Example: db.users.find({age:20},{age:1,_id:0})

Q41. What is $unwind?

Quick answer: Deconstructs array field into separate docs in aggregation.

Example: db.orders.aggregate([{$unwind:'$items'}])

Q42. What is $set in update?

Quick answer: Updates/adds new field in doc.

Example: db.users.updateOne({id:1},{$set:{age:30}})

Q43. What is difference between $inc and $mul?

Quick answer: $inc increments numeric value. $mul multiplies.

Example: db.users.updateOne({_id:1},{$inc:{score:5}})

Q44. What is difference between $push and $addToSet?

Quick answer: $push adds to array, duplicates allowed. $addToSet prevents duplicates.

Example: db.users.updateOne({_id:1},{$addToSet:{tags:'new'}})

Q45. What is aggregation pipeline?

Quick answer: Stages processing docs sequentially ($match,$group,$sort).

Example: db.sales.aggregate([{$match:{status:'A'}},{$group:{_id:'$cust',total:{$sum:'$amt'}}}])

Q46. What is difference between $sort and index sorting?

Quick answer: $sort sorts in pipeline; index sorting uses index order, faster.

Example: db.users.find().sort({age:1})

Q47. How to monitor MongoDB performance?

Quick answer: Use db.serverStatus(), profiler, monitoring tools.

Example: db.setProfilingLevel(2)

Q48. Additional MongoDB concept #48

Quick answer: Explanation of a MongoDB interview topic.

Example: MongoDB command or query example here.

Q49. Additional MongoDB concept #49

Quick answer: Explanation of a MongoDB interview topic.

Example: MongoDB command or query example here.

Q50. Additional MongoDB concept #50

Quick answer: Explanation of a MongoDB interview topic.

Example: MongoDB command or query example here.