

Basic Concepts

1. **MongoDB is classified as what type of database?**

- a) Relational Database
- b) Document Database
- c) Key-Value Store
- d) Columnar Database

Answer: b) Document Database

2. **What is the format used to store documents in MongoDB?**

- a) XML
- b) JSON
- c) BSON
- d) CSV

Answer: c) BSON

3. **Which of the following is the correct way to represent a document in MongoDB?**

- a) {name: "John", age: 30, city: "New York"}
- b) <name>John</name><age>30</age><city>New York</city>
- c) "John",30,"New York"
- d) {"name"="John","age"=30,"city"="New York"}

Answer: a) {name: "John", age: 30, city: "New York"}

CRUD Operations

4. **Which method is used to insert multiple documents in MongoDB?**

- a) insert()
- b) insertOne()
- c) insertMany()
- d) add()

Answer: c) insertMany()

5. **What is the correct syntax to update a document in MongoDB?**

- a) db.collection.update({query}, {update})
- b) db.collection.update({query}, {\$set: {update}})
- c) db.collection.modify({query}, {update})
- d) Both a and b

Answer: d) Both a and b

6. **Which operator is used to increment a field value in MongoDB?**

- a) \$add
- b) \$increase
- c) \$inc

d) \$plus

Answer: c) \$inc

Indexing and Performance

7. What is the purpose of creating indexes in MongoDB?

- a) To reduce database size
- b) To improve query performance
- c) To enforce data types
- d) To create backups

Answer: b) To improve query performance

8. Which index type ensures that fields are unique across documents?

- a) Single field index
- b) Compound index
- c) Unique index
- d) Text index

Answer: c) Unique index

9. What is the default primary key in MongoDB called?

- a) _id
- b) primary_key
- c) id
- d) key

Answer: a) _id

Aggregation

10. Which pipeline stage is used to group documents in MongoDB aggregation?

- a) \$match
- b) \$project
- c) \$group
- d) \$sort

Answer: c) \$group

11. What does the \$lookup stage in aggregation do?

- a) Performs a left outer join
- b) Filters documents
- c) Sorts documents
- d) Limits the number of documents

Answer: a) Performs a left outer join

Replication and Sharding

12. What is the minimum number of nodes required for a MongoDB replica set?

- a) 1

- b) 2
- c) 3
- d) 4

Answer: c) 3

13. What is the purpose of sharding in MongoDB?

- a) To create backups
- b) To improve read performance
- c) To horizontally scale the database
- d) To encrypt data

Answer: c) To horizontally scale the database

14. Which component routes queries to the appropriate shard in a sharded cluster?

- a) mongod
- b) mongos
- c) config server
- d) arbiter

Answer: b) mongos

Security

15. Which authentication mechanism is recommended for MongoDB?

- a) SCRAM-SHA-1
- b) SCRAM-SHA-256
- c) LDAP
- d) Kerberos

Answer: b) SCRAM-SHA-256

16. What is the purpose of MongoDB's role-based access control?

- a) To limit database size
- b) To define what operations users can perform
- c) To improve query performance
- d) To manage indexes

Answer: b) To define what operations users can perform

Advanced Querying & Indexing

1. Which query operator allows you to perform a join-like operation in MongoDB?

- a) \$join
- b) \$lookup
- c) \$merge
- d) \$combine

Answer: b) \$lookup

2. **What is the purpose of the \$expr operator in MongoDB queries?**

- a) To execute JavaScript functions
- b) To use aggregation expressions in query language
- c) To export query results
- d) To explain query execution plans

Answer: b) To use aggregation expressions in query language

3. **Which index property improves performance for sorting operations?**

- a) sparse
- b) unique
- c) background
- d) collation

Answer: d) collation

Aggregation Framework Deep Dive

4. **What does the \$facet stage in aggregation pipelines allow you to do?**

- a) Create multiple pipelines from a single input
- b) Join collections efficiently
- c) Perform full-text searches
- d) Encrypt aggregation results

Answer: a) Create multiple pipelines from a single input

5. **Which operator would you use to reshape documents with computed values in an aggregation pipeline?**

- a) \$project
- b) \$addFields
- c) \$set
- d) Either a or b

Answer: d) Either a or b

6. **What is the purpose of the \$merge stage in aggregation pipelines?**

- a) To combine multiple collections
- b) To write results to a collection with various options
- c) To perform a merge sort on documents
- d) To optimize pipeline execution

Answer: b) To write results to a collection with various options

Performance Optimization

7. **Which tool would you use to analyze query performance in MongoDB?**

- a) mongostat
- b) explain()
- c) db.currentOp()

- d) All of the above

Answer: d) All of the above

8. What is a covered query in MongoDB?

- a) A query that uses a secondary index
- b) A query where all fields are part of an index
- c) A query that returns only indexed fields
- d) Both b and c

Answer: d) Both b and c

9. Which read concern level provides the strongest consistency guarantee?

- a) "local"
- b) "available"
- c) "majority"
- d) "linearizable"

Answer: d) "linearizable"

Advanced Schema Design

10. When would you use the Bucket Pattern in MongoDB schema design?

- a) To store time-series data efficiently
- b) To implement many-to-many relationships
- c) To handle large binary files
- d) To create hierarchical data structures

Answer: a) To store time-series data efficiently

11. What is the primary advantage of the Extended Reference Pattern?

- a) It reduces the need for joins
- b) It improves write performance
- c) It automatically creates indexes
- d) It enforces schema validation

Answer: a) It reduces the need for joins

12. Which schema design pattern would be most appropriate for product catalogs with variable attributes?

- a) Polymorphic Pattern
- b) Attribute Pattern
- c) Subset Pattern
- d) Outlier Pattern

Answer: b) Attribute Pattern

Transactions & Consistency

13. What is the maximum duration for a multi-document transaction in MongoDB?

- a) 30 seconds

- b) 60 seconds
- c) 120 seconds
- d) No strict limit

Answer: b) 60 seconds

14. Which write concern provides the strongest durability guarantee?

- a) {w: 1}
- b) {w: "majority"}
- c) {w: "majority", j: true}
- d) {w: "all"}

Answer: c) {w: "majority", j: true}

15. In a sharded cluster, which operations support multi-document transactions?

- a) Only operations within a single shard
- b) Operations across multiple shards
- c) Both a and b
- d) Neither a nor b

Answer: c) Both a and b

Advanced Cluster Management

16. What is the purpose of the read preference "nearest"?

- a) To read from the primary only
- b) To read from the secondary with lowest replication lag
- c) To read from the node with lowest network latency
- d) To read from a random secondary

Answer: c) To read from the node with lowest network latency

17. Which sharding strategy would be most appropriate for monotonically increasing keys?

- a) Range-based sharding
- b) Hash-based sharding
- c) Zone sharding
- d) Tag-aware sharding

Answer: b) Hash-based sharding

18. What is the role of config servers in a sharded cluster?

- a) To store metadata about chunk distribution
- b) To route queries to appropriate shards
- c) To act as arbiters in elections
- d) To perform background compaction

Answer: a) To store metadata about chunk distribution

Security & Compliance

19. Which encryption method protects data at rest in MongoDB?

- a) TLS/SSL
- b) Field-level encryption
- c) WiredTiger encryption
- d) All of the above

Answer: c) WiredTiger encryption

20. What is the purpose of MongoDB's auditing capability?

- a) To track database operations for compliance
- b) To automatically fix security vulnerabilities
- c) To encrypt all network traffic
- d) To prevent SQL injection attacks

Answer: a) To track database operations for compliance

21. Which role provides the most comprehensive administrative privileges?

- a) dbAdmin
- b) userAdmin
- c) clusterAdmin
- d) root

Answer: d) root

Advanced Features

22. What is the purpose of change streams in MongoDB?

- a) To monitor real-time data changes
- b) To improve replication speed
- c) To change database configurations
- d) To stream data to external systems

Answer: a) To monitor real-time data changes

23. Which MongoDB feature allows you to store and query geospatial data efficiently?

- a) GridFS
- b) 2dsphere indexes
- c) Text indexes
- d) Wildcard indexes

Answer: b) 2dsphere indexes

24. What does the \$graphLookup stage enable in aggregation pipelines?

- a) Graph database capabilities
- b) Recursive searches on hierarchical data
- c) Visualization of query plans
- d) Machine learning predictions

Answer: b) Recursive searches on hierarchical data

25. **When would you use MongoDB's \$out aggregation stage?**

- a) To export data to a file
- b) To write results to a new collection
- c) To output query execution statistics
- d) To terminate an aggregation pipeline

Answer: b) To write results to a new collection

26.1.1 Characteristics of MongoDB

27. **Q1. Which of these is NOT a characteristic of MongoDB?**

- a) Document-oriented storage
- b) Fixed schema requirement
- c) Horizontal scalability
- d) Rich query language

Answer: b) Fixed schema requirement

28. **Q2. MongoDB stores data in which format?**

- a) XML
- b) CSV
- c) BSON
- d) Avro

Answer: c) BSON

29. **Q3. What provides high availability in MongoDB?**

- a) Sharding
- b) Replica sets
- c) GridFS
- d) MapReduce

Answer: b) Replica sets

30. **Q4. Which feature allows MongoDB to process complex data transformations?**

- a) ACID transactions
- b) Aggregation framework
- c) Fixed schemas
- d) Foreign keys

Answer: b) Aggregation framework

31.1.2 Differences Between MongoDB and RDBMS

32. Q1. How does MongoDB primarily handle relationships between data?

- a) Foreign keys
- b) Embedded documents and references
- c) Junction tables
- d) Stored procedures

Answer: b) Embedded documents and references

33. Q2. Which scalability approach is native to MongoDB?

- a) Vertical scaling only
- b) Horizontal scaling via sharding
- c) Partitioning with foreign keys
- d) Read replicas only

Answer: b) Horizontal scaling via sharding

34. Q3. What is the fundamental structural difference between MongoDB and RDBMS?

- a) MongoDB uses tables, RDBMS uses collections
- b) MongoDB uses documents, RDBMS uses tables
- c) MongoDB requires JOIN operations
- d) RDBMS has no schema constraints

Answer: b) MongoDB uses documents, RDBMS uses tables

35. Q4. Which feature was introduced in MongoDB 4.0 to narrow the gap with RDBMS?

- a) Full-text search
- b) Multi-document ACID transactions
- c) Geospatial queries
- d) Fixed schema enforcement

Answer: b) Multi-document ACID transactions

36.1.3 Methods to Access and Administer MongoDB

37. Q1. Which is MongoDB's native interactive interface?

- a) MySQL Workbench
- b) MongoDB Compass
- c) mongosh shell

d) Robo 3T

Answer: c) mongosh shell

38. **Q2. What is the primary configuration file format for MongoDB?**

- a) .ini
- b) .conf (YAML)
- c) .xml
- d) .json

Answer: b) .conf (YAML)

39. **Q3. Which tool provides a graphical interface for MongoDB administration?**

- a) mongod
- b) mongostat
- c) MongoDB Compass
- d) mongos

Answer: c) MongoDB Compass

40. **Q4. How can you monitor real-time MongoDB operations?**

- a) db.currentOp()
- b) SHOW PROCESSLIST
- c) EXPLAIN ANALYZE
- d) TOP command

Answer: a) db.currentOp()

41.1.4 Function of Sharding

42. **Q1. What is the primary purpose of sharding in MongoDB?**

- a) Data backup
- b) Horizontal scaling
- c) Vertical scaling
- d) Query optimization

Answer: b) Horizontal scaling

43. **Q2. Which component routes queries in a sharded cluster?**

- a) Config server
- b) mongod
- c) mongos

- d) Arbiter

Answer: c) mongos

44. Q3. What determines how data is distributed across shards?

- a) Primary key
- b) Shard key
- c) Foreign key
- d) Partition key

Answer: b) Shard key

45. Q4. Which sharding strategy is best for evenly distributing writes?

- a) Range-based
- b) Hash-based
- c) Zone-based
- d) Time-based

Answer: b) Hash-based

46. Bonus Advanced Question:

Q5. What happens when a chunk grows beyond the default 64MB size?

- a) The shard crashes
- b) Automatic chunk splitting occurs
- c) Writes are blocked
- d) A new index is created

Answer: b) Automatic chunk splitting occurs

2.1 Proper and Improper Insert Commands

Q: Which of the following is an *improperly* formed MongoDB insert command?

- a) db.users.insertOne({name: "Alice", age: 25})
 - b) db.users.insert({name: "Bob", age: "30"}) (*Note: age is a string*)
 - c) db.users.insertOne({_id: 1, name: "Charlie"})
 - d) db.users.insert({name: "Dave", age: null})
- Answer:** b) db.users.insert({name: "Bob", age: "30"}) (*Improper if age was intended to be numeric*)

2.2 Update Without Operators (Full Document Replacement)

Q: What happens when you run db.products.replaceOne({_id: 1}, {name: "Laptop", price: 999}) on this document?

Original Document: {_id: 1, name: "Phone", price: 599, stock: 10}

- a) Only name and price are updated.
- b) The document becomes {_id: 1, name: "Laptop", price: 999} (*stock is removed*)
- c) The operation fails due to missing stock.

 **Answer:** b) (*Full replacement removes fields not specified in the update!*)

2.3 Update Using \$set

Q: After running db.users.updateOne({_id: 1}, {\$set: {status: "active"}}, what changes?

Original Document: {_id: 1, name: "Alice", age: 25}

- a) status is added, and age is removed.
- b) The document becomes {_id: 1, name: "Alice", age: 25, status: "active"}.
- c) The operation fails if status didn't exist.

 **Answer:** b) (*\$set adds/updates only the specified field.*)

2.4 Upsert Command

Q: Which command inserts {_id: 2, name: "Bob"} if _id: 2 doesn't exist?

- a) db.users.updateOne({_id: 2}, {name: "Bob"}, {upsert: false})
- b) db.users.updateOne({_id: 2}, {\$set: {name: "Bob"}}, {upsert: true})
- c) db.users.insert({_id: 2, name: "Bob"})

 **Answer:** b) (*Correct upsert: true syntax with \$set.*)

2.5 Update Multiple Documents

Q: How do you add discount: 10 to *all* products with price > 500?

- a) db.products.update({price: {\$gt: 500}}, {\$set: {discount: 10}})
- b) db.products.updateMany({price: {\$gt: 500}}, {\$set: {discount: 10}})
- c) db.products.updateAll({price: {\$gt: 500}}, {discount: 10})

 **Answer:** b) (*updateMany is the correct method.*)

2.6 findAndModify with Concurrency

Q: If two clients run findAndModify simultaneously on the same document, what happens?

- a) Both operations execute in parallel.

- b) The second operation waits for the first to complete (*atomic operation*).
- c) The document is duplicated.

 **Answer:** b) (*MongoDB locks the document during findAndModify.*)

2.7 Delete Document

Q: Which command deletes *one* user with status: "inactive"?

- a) db.users.delete({status: "inactive"})
- b) db.users.deleteOne({status: "inactive"})
- c) db.users.removeOne({status: "inactive"})

 **Answer:** b) (*Correct method is deleteOne.*)

2.8 Equality Constraint Query

Q: How do you find a document where x: 3?

- a) db.collection.find({x: {\$eq: 3}})
- b) db.collection.findOne({3: x})
- c) db.collection.find({x: 3})

 **Answer:** c) (*Simple equality queries don't need \$eq.*)

2.9 Array Equality Constraint

Q: Which documents match db.items.find({tags: ["red", "blue"]})?

- a) {tags: ["red", "blue"]} (*exact match*)
- b) {tags: ["blue", "red"]} (*order matters!*)
- c) {tags: "red"}

 **Answer:** a) (*Array equality requires exact order and elements.*)

2.10 Comparison Operators

Q: Which query finds users with age > 21?

- a) db.users.find({age: {\$gt: 21}})
- b) db.users.find({age: > 21})
- c) db.users.find("age > 21")

 **Answer:** a) (*Correct \$gt syntax.*)

2.11 \$in Operator

Q: How do you find products with category in ["electronics", "furniture"]?

- a) db.products.find({category: {\$in: ["electronics", "furniture"]}})
- b) db.products.find({category: ["electronics", "furniture"]})

 **Answer:** a) (*\$in checks for value in array.*)

2.12 \$elemMatch

Q: Which query matches documents where *one array element* satisfies both score > 80 and subject: "math"?

- a) db.scores.find({results: {\$elemMatch: {score: {\$gt: 80}, subject: "math"}}})
- b) db.scores.find({results.score > 80, results.subject: "math"})

 **Answer:** a) (*\$elemMatch ensures both conditions apply to the same array element.*)

2.13 Logical Operators

Q: Which query finds users with age < 18 **OR** status: "inactive"?

- a) db.users.find({\$or: [{age: {\$lt: 18}}, {status: "inactive"}]})
- b) db.users.find({age < 18, status: "inactive"})

 **Answer:** a) (*Correct \$or syntax.*)

2.14 Sort + Limit

Q: What does db.products.find().sort({price: -1}).limit(2) return?

- a) The 2 cheapest products.
- b) The 2 most expensive products (*descending sort*).

 **Answer:** b) (-1 sorts descending.)

2.15 Incorrect Projection

Q: Which projection is *invalid*?

- a) db.users.find({}, {name: 1, _id: 0})
- b) db.users.find({}, {name: 1, age: 1, _id: 1})
- c) db.users.find({}, {name: "true", age: "true"}) (*Must use 1/0, not strings!*)

 **Answer:** c) (*Projection requires 1 (include) or 0 (exclude).*)

2.16 Count Documents

Q: How do you count users with status: "active"?

- a) db.users.count({status: "active"}) (*Deprecated*)
- b) db.users.find({status: "active"}).count() (*Deprecated*)
- c) db.users.countDocuments({status: "active"}) (*Modern method*)

 **Answer:** c) (*Prefer countDocuments() over legacy methods.*)

2.17 Aggregation (\$match + \$group)

Q: What does this aggregation output?

javascript

Copy

Download

```
db.orders.aggregate([  
  {$match: {status: "completed"}},  
  {$group: {_id: "$customer", total: {$sum: "$amount"}}}  
)
```

- a) Total amount per customer for *completed* orders.
- b) All orders grouped by status.

 **Answer:** a) (*\$match filters, \$group aggregates.*)

Bonus: Atomicity Question

Q: Which operation is *atomic* at the document level?

- a) updateMany()
- b) findAndModify()

 **Answer:** b) (*Only findAndModify is atomic for a single document.*)

4.1 Starting mongod

Q1: What is the correct command to start a MongoDB server with default settings?

- a) start mongodb
- b) mongod --start
- c) mongod
- d) mongo --start-server

 **Answer:** c) mongod

Q2: Which parameter would you add to specify a custom data directory when starting mongod?

- a) --dbpath /data/db
- b) --datadir /data/db
- c) --storage /data/db
- d) --directory /data/db

Answer: a) --dbpath /data/db

4.2 Checking mongod Status

Q3: How can you verify if mongod is running on Linux?

- a) ps aux | grep mongod
- b) mongod --status
- c) service mongod check
- d) db.serverStatus()

Answer: a) ps aux | grep mongod

Q4: Which command in mongosh checks if you're connected to a running server?

- a) db.isConnected()
- b) db.serverStatus()
- c) db.runCommand({ping: 1})
- d) show connections

Answer: c) db.runCommand({ping: 1})

4.3 Stopping mongod

Q5: What is the proper way to shut down a mongod instance cleanly?

- a) mongod --stop
- b) db.shutdownServer() in mongosh
- c) kill -9 <mongod_pid>
- d) service mongod kill

Answer: b) db.shutdownServer() in mongosh

Q6: Which signal should you send for a graceful shutdown?

- a) SIGKILL (9)
- b) SIGTERM (15)
- c) SIGHUP (1)
- d) SIGINT (2)

Answer: b) SIGTERM (15)

4.4 Adding Parameters

Q7: How would you start mongod with authentication enabled?

- a) mongod --enable-auth

- b) mongod --auth
- c) mongod --security=on
- d) mongod --with-authentication

 **Answer:** b) mongod --auth

Q8: Where can you persistently store mongod parameters?

- a) In the MongoDB dashboard
- b) In a configuration file (usually /etc/mongod.conf)
- c) In the admin database
- d) In the local.system.params collection

 **Answer:** b) In a configuration file

4.5 Checking Parameter Values

Q9: How can you view all current runtime configuration options?

- a) db.getConfig()
- b) db.adminCommand({getParameter: '*'})
- c) mongod --show-parameters
- d) db.serverParameters()

 **Answer:** b) db.adminCommand({getParameter: '*'})

Q10: Which command shows the storage engine configuration?

- a) db.serverStatus().storageEngine
- b) db.getStorageEngines()
- c) show engine status
- d) db.stats().engine

 **Answer:** a) db.serverStatus().storageEngine

4.6 Using mongosh

Q11: How do you connect to a MongoDB server on port 27018 using mongosh?

- a) mongosh --port 27018
- b) mongo --port 27018
- c) connect mongodb://localhost:27018
- d) mongosh localhost:27018

 **Answer:** a) mongosh --port 27018

Q12: Which mongosh command displays available databases?

- a) show dbs
- b) db.listDatabases()
- c) list databases
- d) db.showDatabases()

 **Answer:** a) show dbs

4.7 Replica Set Management

Q13: How do you add a new node to a replica set?

- a) rs.addNode("hostname:port")
- b) rs.add("hostname:port")
- c) db.addReplica("hostname:port")
- d) replicaSet.addMember("hostname:port")

 **Answer:** b) rs.add("hostname:port")

Q14: What command removes a node from a replica set?

- a) rs.kick("hostname:port")
- b) rs.remove("hostname:port")
- c) db.removeNode("hostname:port")
- d) replicaSet.remove("hostname:port")

 **Answer:** b) rs.remove("hostname:port")

Q15: How do you check the current replica set configuration?

- a) rs.status()
- b) rs.config()
- c) db.getReplicaSetStatus()
- d) show replica config

 **Answer:** b) rs.config()

Q16: What is the correct sequence to initialize a new replica set?

1. Start mongod instances with --replSet parameter
2. Connect to the primary member
3. Run rs.initiate()
4. Add other members with rs.add()
 - a) 1-2-3-4
 - b) 1-3-2-4
 - c) 2-1-3-4
 - d) 3-1-2-4

 **Answer:** a) 1-2-3-4

5.1 Common Alerts

Q1: What does a "Replication lag" alert typically indicate?

- a) The primary node is overloaded
- b) Secondaries are falling behind the primary in applying operations
- c) The oplog is full

- d) Network latency between clients and MongoDB

Answer: b) Secondaries are falling behind the primary

Q2: A "High resident memory usage" alert suggests:

- a) The server is running out of disk space
- b) MongoDB is using too much RAM for working set
- c) Too many connections are open
- d) The WiredTiger cache is improperly sized

Answer: b) High RAM usage for working set

5.2 Running currentOp

Q3: Which command shows all currently executing operations?

- a) db.currentOp()
- b) db.adminCommand({currentOp: 1})
- c) show current operations
- d) Both a and b

Answer: d) Both a and b (equivalent commands)

Q4: To filter only active operations running longer than 5 seconds:

- a) db.currentOp({"secs_running": {\$gt: 5}})
- b) db.currentOp({"active": true, "secs_running": {\$gt: 5}})
- c) db.adminCommand({currentOp: 1, filter: {op: "query", secs_running: {\$gt: 5}}})

Answer: b) Correct filter syntax

5.3 Active Operations Monitoring

Q5: Which field in currentOp output indicates a query is blocking other operations?

- a) "waitingForLock": true
- b) "lockStats": {"waiting": true}
- c) "blocking": true
- d) "active": true

Answer: a) "waitingForLock": true

Q6: How would you identify slow queries (e.g., >100ms) in real-time?

- a) db.currentOp({"secs_running": {\$gt: 0.1}})
- b) MongoDB Profiler
- c) db.setProfilingLevel(1, 100)
- d) All of the above

Answer: d) All are valid approaches

5.4 Storage Monitoring

Q7: Which command shows storage utilization per database?

- a) db.stats()
- b) db.adminCommand({listDatabases: 1})
- c) show storage
- d) db.serverStatus().storage

 **Answer:** a) db.stats()

Q8: To check WiredTiger cache usage:

- a) db.serverStatus().wiredTiger.cache
- b) db.adminCommand({cacheStats: 1})
- c) db.storageStats()

 **Answer:** a) Correct WiredTiger cache path

5.5 Storage Capacity Warnings

Q9: Which log warning indicates critically low disk space?

- a) "Disk space is almost full"
- b) "Storage engine is in read-only mode"
- c) "No space left on device"
- d) All of the above

 **Answer:** d) All are severe indicators

Q10: What happens when MongoDB runs out of disk space?

- a) Automatically deletes old data
- b) Switches to read-only mode
- c) Shuts down gracefully
- d) Continues operating normally

 **Answer:** b) Read-only mode to prevent corruption

5.6 Connection Monitoring

Q11: Which command shows current connections?

- a) db.serverStatus().connections
- b) db.adminCommand({connPoolStats: 1})
- c) netstat -an | grep 27017
- d) Both a and c

 **Answer:** d) MongoDB + OS-level checks

Q12: Given this output, how many connections are *active*?

json

Copy

Download

```
{  
  "current": 25,  
  "available": 512,  
  "totalCreated": 1200  
}
```

- a) 25
- b) 512
- c) 1200
- d) 537 (512 + 25)

 **Answer:** a) current shows active connections

Bonus: OOM Killer Scenario

Q13: If MongoDB suddenly crashes and /var/log/messages shows "Out of memory: Kill process", what likely happened?

- a) Disk failure
- b) Linux OOM killer terminated mongod
- c) Replica set election
- d) WiredTiger cache overflow

 **Answer:** b) OOM killer action

Key Monitoring Tools

Q14: Which tool provides real-time cluster metrics?

- a) MongoDB Atlas
- b) mongostat
- c) db.currentOp()
- d) All of the above

 **Answer:** d) All are valid

7.1 Purpose of Replication

Q1: What is the primary purpose of MongoDB replication?

- a) To increase storage capacity
- b) To provide high availability and data redundancy
- c) To improve query performance through parallel processing
- d) To enable sharding across multiple servers

 **Answer:** b) To provide high availability and data redundancy

7.2 Purpose of the Primary

Q2: In a replica set, what is the role of the primary node?

- a) To handle all read operations
- b) To receive all write operations and record changes in the oplog
- c) To act as a backup for secondary nodes
- d) To route queries to the appropriate shard

 **Answer:** b) To receive all write operations and record changes in the oplog

7.3 Purpose of the Secondary

Q3: Which of these is NOT a function of secondary nodes?

- a) Maintaining a copy of the primary's data
- b) Participating in elections
- c) Processing write operations from clients
- d) Serving read operations when configured

 **Answer:** c) Processing write operations from clients (only primary handles writes)

7.4 Read Preference Uses

Q4: When would you use readPreference: "secondary"?

- a) When you need the most up-to-date data
- b) When you want to distribute read load from the primary
- c) When performing write operations
- d) When connecting to a standalone MongoDB instance

 **Answer:** b) When you want to distribute read load from the primary

7.5 rs.status() Output

Q5: What information can you find in rs.status() output?

- a) Current primary and secondary members
- b) Replication lag between nodes
- c) Member health and state
- d) All of the above

 **Answer:** d) All of the above

7.6 Write Concern Purpose

Q6: Why would you set writeConcern: {w: "majority"}?

- a) To ensure data is written to memory only
- b) To guarantee writes are durable across multiple nodes
- c) To disable journaling for better performance
- d) To limit the number of write operations

Answer: b) To guarantee writes are durable across multiple nodes

7.7 Election Purpose

Q7: What triggers a replica set election?

- a) When a secondary node receives a write operation
- b) When the primary becomes unavailable
- c) When a new index is created
- d) When oplog size is exceeded

Answer: b) When the primary becomes unavailable

7.8 Primary Failure Scenario

Q8: If a primary fails, what happens next?

- a) All secondaries become read-only permanently
- b) The replica set becomes unavailable until manual intervention
- c) The secondaries hold an election to select a new primary
- d) MongoDB automatically switches to standalone mode

Answer: c) The secondaries hold an election to select a new primary

7.9 Oplog Purpose

Q9: What is the function of the oplog?

- a) To store user credentials for authentication
- b) To record all write operations for replication
- c) To optimize query performance
- d) To manage sharding metadata

Answer: b) To record all write operations for replication

Q10: What is the minimum number of nodes required for a replica set to maintain availability during a single failure?

- a) 1
- b) 2
- c) 3
- d) 4

Answer: c) 3 (allows for one failure while maintaining majority)

Q11: Which read preference mode would you use for geographically local reads in a globally distributed replica set?

- a) "primary"
- b) "secondaryPreferred"
- c) "nearest"
- d) "primaryPreferred"

 **Answer:** c) "nearest" (lowest latency)

Q12: What happens if a secondary node's oplog cannot keep up with changes from the primary?

- a) The secondary automatically resyncs
- b) The secondary enters "ROLLBACK" state
- c) The primary stops accepting writes
- d) Replication continues from the oldest oplog entry

 **Answer:** a) The secondary automatically resyncs (in MongoDB 4.0+)

1. MongoDB Backup Methods

Q1: Which of these is NOT a valid MongoDB backup method?

- a) mongodump
- b) Filesystem snapshots
- c) rsync
- d) Oplog tailing

 **Answer:** c) rsync (*Not a dedicated MongoDB backup tool*)

2. mongodump Features

Q2: What does mongodump --oplog do?

- a) Creates a backup of the oplog only
- b) Includes oplog entries for point-in-time recovery
- c) Backups only the most recent operations
- d) Disables oplog during backup

 **Answer:** b) Includes oplog entries for point-in-time recovery

3. Filesystem Snapshots

Q3: When taking a filesystem snapshot for MongoDB, what must you ensure?

- a) The mongod process is stopped
- b) The journal files are excluded
- c) The data files are in a consistent state
- d) The oplog is truncated first

 **Answer:** c) The data files are in a consistent state (*Journaling ensures this*)

4. Point-in-Time Recovery

Q4: Which component enables point-in-time recovery in MongoDB?

- a) Journal files
- b) Oplog
- c) WiredTiger snapshots
- d) mongorestore --timestamp

 **Answer:** b) Oplog (*Records all changes for replay*)

5. mongorestore Limitations

Q5: What does mongorestore NOT preserve by default?

- a) Indexes
- b) Document _id fields
- c) Original timestamps of documents
- d) Collection options

 **Answer:** c) Original timestamps of documents (*Unless using --oplogReplay*)

6. Backup Authentication

Q6: How do you authenticate mongodump against a secured MongoDB deployment?

- a) --username and --password flags
- b) --authenticationDatabase parameter
- c) Connection string with credentials
- d) All of the above

 **Answer:** d) All of the above

7. Journaling Purpose

Q7: What is the primary role of MongoDB's journal?

- a) To enable replication
- b) To recover from crashes without data loss
- c) To optimize query performance
- d) To track slow operations

 **Answer:** b) Crash recovery (*Ensures write durability*)

8. Hot Backups

Q8: Which backup method allows backing up a running MongoDB instance?

- a) mongodump with --forceTableScan
- b) LVM snapshots with journaling enabled
- c) Copying data files after stopping mongod
- d) Both a and b

 **Answer:** d) Both a and b (*Hot backup methods*)

9. Oplog Size Calculation

Q9: What determines the minimum oplog size needed for backups?

- a) Total database size
- b) The time window of changes to retain
- c) Number of indexes
- d) WiredTiger cache size

 **Answer:** b) Time window of changes to retain (*At least 72 hours recommended*)

10. Disaster Recovery

Q10: For full disaster recovery, what must you back up besides data files?

- a) mongod process logs
- b) Configuration files
- c) Keyfile for replica set authentication
- d) All of the above

 **Answer:** d) All of the above

11. Encrypted Backups

Q11: How can you ensure mongodump backups are encrypted?

- a) Use --encrypt flag
- b) Enable WiredTiger encryption first
- c) Dump to an encrypted filesystem
- d) Both b and c

 **Answer:** d) Both b and c (*No native mongodump encryption*)

12. Partial Backups

Q12: Which mongodump option backs up only specific collections?

- a) --includeCollections
- b) --filter

- c) --db and --collection
 - d) --query
-  **Answer:** c) --db and --collection
-

13. Cloud Backup Solutions

Q13: Which MongoDB service offers built-in automated backups?

- a) MongoDB Atlas
- b) MongoDB Ops Manager
- c) MongoDB Cloud Manager
- d) All of the above

 **Answer:** d) All of the above

14. Restore Strategies

Q14: When restoring a sharded cluster, what's critical to do first?

- a) Restore config servers
- b) Restore the largest shard
- c) Disable balancer
- d) Clear all existing data

 **Answer:** a) Restore config servers (*Metadata must be restored first*)

15. Backup Monitoring

Q15: How can you verify a backup's integrity?

- a) Check backup file size
- b) Run mongorestore --dryRun
- c) Validate checksums
- d) Restore to a test environment

 **Answer:** d) Restore to a test environment (*Only definitive method*)