MongoDB

1. **Theory**
2. SQL(relational) v s
3. NoSQL ()
4. What is MongoDB?
5. Run on JS Engine
6. How does mongoDB work?
7. Non-relational Document based
8. Advantage and Disadvantages
9. BSON
10. MongoDB Structure
11. MongoDB architecture
12. JSON vs BSON
13. MongoDB shell
14. CRUD Operations
15. Cursor, Iterate a Cursor
16. Time to Leave
17. Maximum Document Size : 16Mb
18. **Storage engines**
    a. **types**
        i. WiredTiger
        ii. ger engine
        iii. In-memory engine
        iv. MMAPv1
    b. GridFS
    c. Journal
19. **Data types in MongoDB (BSON)**
    a. ObjectId
        i. timestamp
        ii. random value
        iii. incrementing counter
    b. String
    c. Int, longInt, Double
    d. Array, Object
    e. Boolean
    f. Date
    g. Decimal128
    h. Regex
    i. Javascript
        i. with scope
        ii. without scope
    j. MinKey, MaxKey
    k. Binary data
20. Cursor
    a. cursor methods
    b. - toArray
    c. - forEach
    d. cursor.allowPartialResults()
21. **Collection**
    a. db
    b. db.createCollection(collectionName)
    c. show collections
    d. renaming Collection
22. **Documents**
    a. adding new Documents
    b. Nested Documents
        i. advantage
23. **Inserting Document**
24. Insert One and Many
25. what are the additional methods used for inserting
26. **Finding / Querying**
    a. find()+
        i. iterate (it)
        ii. pretty()
    b. findOne({ *filter* })
    c. finding In nested Array
        i. "*field.field*"
        ii. match
        iii. exact match
        iv. multiple match
    d. Array
        i. finding in specific order
        ii. without regard to order
        iii. query by array index
        iv. query by array length
    e. **Projection**
        i. explicitly include fields
    f. Null, $type: 10, $exists
27. **Filtering**
    a. find(*filter*)
    b. find(*{filter}, {fieldsToGet}*)
28. **Method Chaining**
    a. count()

- b. limit()
- c. sort(1 or -1) ,
- d. skip()

29. **Operators** (denoted by $)
    - a. {$gt: number} $gte
    - b. $lt, $lte
    - c. $eq, $ne
    - d. $or $and $not
    - e. $in: [1,2,3], $nin: [1,2]
    - f. $all
    - g. $set, $unset
    - h. $addToSet
    - i. **$elemMatch**
    - j. $slice
    - k. $size
    - l. $inc: 1, $inc: -1
    - m. $pull, $push
    - n. $each [1, 2]
    - o. $eq, $ne
    - p. $currentDate
    - q. $exists
    - r. **$expr**
    - s. **$cond**
    - t. $rename
    - u. $min, $max
    - v. $mul
    - w. $ifNull
    - x. $let
    - y. **Array Operator**
        - i. $push
        - ii. $each
        - iii. $pull
        - iv. $pullAll
        - v. $pop
        - vi. $elemMatch
30. **Deleting**
    - a. deleteOne({ *field:value* })
    - b. deleteMany()
    - c. remove()
    - d. delete vs remove
31. **Updating**
    - a. updateOne({*whichObject*} , {$set: {*field: value, field: value*} })
    - b. **Operators**
        - i. $set
        - ii. $unset
        - iii. $rename
    - c. updateMany()
    - d. replaceOne()
    - e. incrementing & decrementing
    - f. adding and remove from array
    - g. upsert
    - h. update() vs updateOne()
    - i. updateOne vs replaceOne
32. **bulkWrite()**
    - a. ordered: false
    - b. ordered vs unordered
    - c. advantages and disadvantages
33. **Commands**
    - a. mongosh
    - b. db
    - c. show dbs
    - d. db.stats
34. **Aggregation**
    - a. How does it work
    - b. advantages
    - c. types of aggregation
    - d. distinct
    - e. **Aggregate stages**
        - i. $addFields
        - ii. $match
        - iii. $group
            1. grouping by
            2. -nested field
            3. -multiple field
        - iv. $sort
        - v. $set
        - vi. $count
        - vii. - other ways to count
        - viii. - client and server side counting
        - ix. $limit, $skip
        - x. $merge
        - xi. $out
        - xii. $project
        - xiii. $lookup

- xiv. $unwind
- xv. $facet
- xvi. $fill
- xvii. $bucket
  - 1. $bucketAuto
- xviii. $densify
- xix. $redact
- xx. $search
- xxi. allowDiskUse: true

f. “$name” vs ”name”

g. **Accumulator Operators**
- i. $sum, $avg, $max, $min

h. **Unary Operators**
- i. $type, $lt $gt $or $and $multiply

i. **Aggregation Pipeline**
- i. How does aggregation pipeline work?
- ii. memory limit : 100mb
  - 1. spill to disk

j. Batch sizing

k. Iterator Size

l. Query routing

m. **Map Reduce**
- i. for what is it used?
- ii. find sum, avg

35. **Indexes**
    - a. pros and cons of Indexes
    - b. createIndex({ *filed: value* })
    - c. options when creating Index
      - i. background: true
      - ii. unique: true
      - iii. name: “<*indexName*>”
    - d. getIndex()
    - e. dropIndex(), dropIndexes
    - f. reIndex()
    - g. rename Index
    - h. hiding index
    - i. covered query
    - j. **Types of Indexes**
      - i. Single Field Index
      - ii. Compound Index
      - iii. Multikey Index
      - iv. Text Index
      - v. Geospatial, Hashed, Clustered Index
      - vi. Covered query
      - vii.
36. **Schema**
    - a. pros and cons of using schema
    - b. optional schema
    - c. validation action
37. **Relationships**
    - a. embedding
    - b. referencing
    - c. one-to-one
    - d. one-to-many
    - e. one-to-squillions
    - f. many-to-many
38. **Replication**
    - a. replica set
    - b. advantage and disadvantages of replication
    - c. **Replication Architecture**
      - i. primary and secondary nodes
      - ii. arbiter
      - iii. process of election
      - iv. heartbeat
    - d. Process of Election
    - e. Replication lag
    - f. operation log (oplog)
    - g. **Types of replication**
      - i. Asynchronous Replication
      - ii. Synchronous Replication
      - iii. Majority Commit
      - iv. etc…
39. **Sharding**
    - a. advantages and disadvantages
    - b. **Sharding Architecture**
      - i. What is Mongos/Router
      - ii. Config Server
    - c. **Types of sharding**

        i. Hashed sharding
        ii. Ranged sharding
        iii. Zone Sharding
    d. **Shard key**
        i. shard hotspots
        ii. normal shard key
        iii. hashed shard key
    e. Vertical and horizontal scaling
    f. Zones
    g. mongos
    h. auto balancer
    i. scatter-gather

40. **Cluster**
    a. types of cluster
    b. config servers
41. **Data Modeling**
    a. embedded data model
    b. reference data model
    c. linking vs embedding
42. **Transactions**
    a. How to do transaction
        i. **Session**
        ii. startTransaction
        iii. abortTransaction
        iv. commitTransaction
    b. ACID Transaction
    c. A- Atomicity
    d. C- Consistency
    e. I - Isolation
    f. D - Durability
43. Create view in Mongodb
44. CAP Theorem
    a. theorem
    b. C- Consistency
    c. A - Availability
    d. P - Particle tolerance
45. **Isolation levels**
    a. Read Concerns
    b. - local
    c. - maojiry
    d. - available
    e. Write Concerns
    f. - w:1 (Acknowledged)
    g. - w:0 (Unacknowledged)
    h. - majority
    i. - all
    j. - journaled
46. **VS**
    a. $or vs $in
    b. $all vs $in
    c. $elemMatch vs $in
    d. drop() vs remove()
    e. findAndModify() vs findOneAndUpdate()
    f. Primary key vs secondary key
    g. join vs lookup
    h. dot notation vs nested form
    i. $currentDate vs $$NOW
    j. delete() vs remove()
    k. bulkWrite vs InsertMany
    l. replace vs update
    m. shard vs node vs cluster
    n. Aggregation Pipeline vs Map Reduce
    o. vertical scalability vs horizontal scalability
    p. load balancer vs sharding
    q. odm vs driver
    r. stage operator vs accumulator operator
    s. normal shard key vs hashed shard key
    t. aggregate([$count:”tota”]) vs find({}).count()
    u. replication vs replica set
    v. transaction vs query
    w. scaling up vs scaling down vs scaling out?
    x. config servers vs mongos
    y. load balancer vs auto balancer
    z. countdocument vs count
47. What is a MongoDB driver?
48. Capped collection and it’s advantages
49. Profiler
50. Explain

51. Soft deleting
52. **Interview Question**
53. What to do when your quireing becomes slow?
54. What to do when your files are getting very big?
55. How to condense large volumes of data?
56. How to search for text in MongoDB?
57. How does MongoDB schema change?
58. How can we Backup and Restore in MongoDB?
59. What are the pros and cons of Normalising Data in MongoDB
60. **Good to Know**
61. Atomicity
62. Type Bracketing
63. Dot Notation
64. Cursor behaviour
65. Aggregation Pipeline
66. Retryable Writes and Reads
67. MongoDB CRUD Concepts
68. B-Tree
69. ACID compliance
70. Mongoose
71. Network Components
    a. load balancer
    b. firewall
72. **CAP Theorem**
    a. consistency
    b. availability
    c. partition tolerance
73. Firewall
74. **Mongo Utilities**
    a. mongoexport
    b. mongoimport
    c. mongodump
    d. mongorestore
    e. mongostat
    f. mongotop
    g. mongooplog
75. Clustered collections
76. WAL