1. What is meant by non relational database?

A non-relational database just stores data without explicit and structured mechanisms to link data from different tables (or buckets) to one another.

2. Compare non relational and relational database?

Attached image

3. When to go for relational database? Non-relational database?

**Relational:**

1. Data transactions secure and making complex queries to acquire information

2. Not having massive growth

**Non-relational:**

1. Strong large amounts of data with little structure

2. growing at a rapid pace

3. for it's scalability and flexibility

4. Why MongoDB is popular ?

- High performance, high availability and automatic scaling,

- MongoDB is extremely simple to install and implement

- MongoDB uses JSON or BSON documents to store data

5. What is the architecture of MongoDb?

Attached image

6. What is the Storage Engine in MongoDB?

- The storage engine is the components of the database

- Responsible for managing how data is stored, both in memory and on disk

- Supports multiple storage engines, as different engines perform better for specific workloads

- mongod --storageEngine wiredTiger

WiredTiger Storage Engine (Default)

WiredTiger is the default storage engine starting in MongoDB 3.2. It is well-suited for most workloads and is recommended for new deployments. WiredTiger provides a document-level concurrency model, checkpointing, and compression, among other features.

In MongoDB Enterprise, WiredTiger also supports Encryption at Rest. See Encrypted Storage Engine.

➤ In-Memory Storage Engine

In-Memory Storage Engine is available in MongoDB Enterprise. Rather than storing documents on-disk, it retains them in-memory for more predictable data latencies.

7. How to start MongoDb?

mongod

8. How to configure mongodb with a file?

mongod --config /etc/mongod.conf\

- logFile:

- storage:

- systemLog:

9. Where data saved? Which format?

- BSON/JSON

- Where /data/db to path or can be changed by configuring using dbPath

10. How to connect to MongoDB?

- mongo with default options

- mongo "mongodb://localhost:27017/employee"

11. Data types available in MongoDb?

- String:

- Integer:

- Boolean:

- Double:

- Min/Max keys:

- Arrays:

- Timestamp: ctimestamp

- Object:

- Null:

- Symbol:

- Date:

- Object ID: This is data type is used to store the document's ID

- Binary data: This data type is used to store binary data.

- Code: This data type is used to store JS code into the document

- Regular expression: This data type is used to store regular

12. What is a collection? How to create them?

db.createCollection("employee");

db.createCollection("log", {capped: true, size: 5242880, max: 5000})

14. Retrieve documents from a collection?

db.employees.findOne({ name: "ASD"});

db.employees.find({}).pretty();

db.employees.find({ position\_held: "Manager"}).pretty();

15. How to insert documents in MongoDB?

Single document

db.employee.insertOne({

"key": 7,

"name": "ASD",

"data\_of\_birth": "10/10/2010",

"position\_held": "Manager"

})

Many documents

db.employeed.insertMany([

{}

])

16. What do you meant by ObjectID in documents?

- The \_id field is always the 1st field in the documents

- unique \_id field that acts as a primary key

If an inserted document omits the \_id filed, the MongoDB driver automatically generates an ObjectID for the \_id field

- 12 byte binary BSON type that contain any 12 bytes you want

- MongoDB drivers and the server will generate them using a default Algorithm

17. Import collections from a json?

mongoimport "tv-shows.json" -d tv -c shows --jsonArray --drop

18. Drop a collection?

db.shows.drop()

19. Access help commands in MongoDB?

db.help()

db.employees.help()

20. How to update documents in MongoDB?

db.employees.updateOne({

{name: "ASD"},

{$set:{date\_of\_birth: "30/10/2010"}}

})

db.employees.updateMany(

{position\_held: "Manager"}, {$set: {date\_of\_birth: "15/10/2010" }}

)

21. What is a projection?

db.employees.find({}, { name: 1, id: 0})

22. What is a cursor object?

23. Limit documents retrieved?

db.employees.find({}).sort({name: 1 }).limit(2).pretty();

24. Find entries with a specific type?

db.employees.find({ key: {$type: "number"}}).pretty();

25. Sort documents?

db.employees.find({}).sort({name: 1}).pretty();

db.employees.find({}).sort({name: -1}).pretty();

26. How to stop the mongo service?

net stop MongoDB