1. What's MongoDB?

Is a popular open-source NoSQL document-oriented database (key-value pairs) that stores data in flexible, JSON-like documents and provides high performance and scalability also it can handle large amount of unstructured or semi-structured data.

2. Important Features?
    - Schema-less database.
    - No complex joins.
    - Faster access to data.
    - Features like aggregation, sharding and replication.
    - Cross-platform and document-based.

3. What's Mongo Shell or MongoSh?

A JavaScript and Node JS REPL (Read Eval Print Loop) environment for interacting with MongoDB deployments locally or on another remote host.

4. How does MongoDB store data?

Being document-based database, MongoDB stores documents in BSON format.

5. What's BSON?

BSON stands for Binary JSON that is binary encoded serialization of JSON-like documents. This binary format leads to more efficient storage and transmission, especially with large datasets. This also provides additional data types such as date, regular expressions and etc.

6. MongoDB is called schema-less database. If yes, how to create schema?

MongoDB is often referred to as a "schema-less" or "schema-flexible" database because, unlike traditional relational databases, it does not enforce a rigid, predefined schema for the documents stored in a collection. MongoDB will dynamically create the collection and add the documents, inferring the structure from the inserted data.

7. What's namespace?

Concatenation of database name and collection name (db_name.collection_name).

8. What's denormalization?

Denormalization is the process of adding redundant data to improve read performance. It's useful when frequent read operations are more critical than write operations.

9. CRUD?
    - Create – db.collection.insert({ username: 'John Doe', age: 30 });
    - Read – db.collection.find({ age: { $gte: 25 } });
    - Update – db.collection.update({ username: 'john_doe' }, { $set: { age: 31 } });
    - Remove – db.collection.remove({ username: ‘inactive_user’ });

10. Default index created for new collection?

_id (primary key).

11. Why’s indexing important?

Indexing improves query performance by allowing the database to locate and retrieve data more quickly.

12. What’s compound index?

compound index involves multiple fields and can be used to support queries with multiple conditions.

13. How to create and drop collections?

db.createCollection(‘myCollection’) & db.collection.drop()

14. What’s Sharding?

Sharding is a method used to distribute data across multiple machines to improve the scalability and performance of a MongoDB database. It allows MongoDB to horizontally partition a collection (divide it into smaller pieces called shards) and store those pieces on different servers or clusters. Each shard is an independent database that stores a subset of the data.

15. How to store images, videos or other large files?

Using the GridFS specification. GridFS is a way to store and retrieve large files that exceed the BSON document size limit of 16 megabytes.

16. Commands to create backup and restore of data?
    - mongodump --host localhost --port 27017 --username myuser --password mypass --db mydatabase --out /path/to/backup
    - mongorestore --host localhost --port 27017 --username myuser --password mypass --db mydatabase --dir /path/to/backup

17. What's Covered query?

A covered query in MongoDB is a type of query where all the fields used in the query are covered by an index, and the index itself provides all the data needed to fulfill the query without the need to look up the actual documents in the collection.

```
{
  "_id": ObjectId("..."),
  "employee_id": 101,
  "name": "Alice",
  "department": "Engineering",
  "salary": 75000,
  "hire_date": ISODate("2022-01-01T00:00:00.000Z")
}
```

```
// Create the index
db.employees.createIndex({ department: 1, hire_date: 1 });

// Covered query example
const result = db.employees.find(
  { department: "Engineering", hire_date: { $gte: ISODate("2022-01-01") } },
  { _id: 0, name: 1, salary: 1 }
);
```

Since the index covers both the query conditions and the fields to be returned, MongoDB can fulfill the query by scanning the index alone.

18. What's $elemMatch operator?

Used to match an element within an array satisfying multiple criteria.

19. What's aggregation framework?

The Aggregation Framework is a powerful tool for data transformation and analysis in MongoDB, providing operations like $match, $group, $project, etc.

20. Explain $lookup stage in aggregation framework?

It performs a left outer join to another collection, combining documents from two collections.

21. What's replication?

Replication involves maintaining multiple copies of data for fault tolerance, high availability, and read scalability.

22. Does MongoDB support transactions?

Yes, MongoDB supports multi-document transactions starting from version 4.0.

23. What's capped collections?

Capped collections are fixed-size collections, and insert and retrieve data based on the insertion order. If a collection's space is full, the oldest records will be overwritten by the new documents in the collection.

24. Purpose of map-reduce command?

- The Map function emits the key-value pair specified.
- The Reduce function combines the key value pair and returns the aggregation result.