



Top 20 MongoDB Interview Questions and Answers

Introduction

Many developers work with MongoDB in web projects today. It is a database that stores data as documents. This format allows applications to manage data with more flexibility. Because of this design, many modern systems use MongoDB in the backend.

During interviews for backend or full stack roles, recruiters often ask questions about this database. Topics usually include collections, queries, and indexing. Reviewing **MongoDB Interview Questions and Answers**

before an interview helps you refresh these ideas. It also helps you explain how MongoDB is used in real applications during technical discussions.

List of MongoDB Interview Questions for Freshers

1. What is MongoDB?
2. In MongoDB, what is a document?
3. Provide the features of MongoDB
4. Describe Namespace
5. What type of NoSQL database is MongoDB?
6. Describe MongoDB's indexes.
7. Explain the importance of the covered query
8. Define replica set.
9. Describe the main and backup replica sets.
10. How do you handle transactions or locking in MongoDB?

MongoDB Technical Interview Questions and Answers for Freshers

1. What is MongoDB?

MongoDB is a cross-platform, document-oriented database system that is open source. MongoDB is a NoSQL database solution that uses documents that resemble JSON and may have optional schemas.

2. In MongoDB, what is a document?

An ordered collection of keys with corresponding values is called a document in MongoDB. It is shown as a dictionary, hash, or map. Documents are represented as objects in JavaScript:

```
{ "greeting": "Hello world!" }
```

Multiple key/value pairs will be present in complex texts.

```
{ "greeting": "Hello world!", "views": 3 }
```

3. Provide the features of MongoDB

The key features of MongoDB are as follows:

- A conforming data model in document format.
- A very scalable and flexible database.
- Faster than conventional databases.
- Clear and concise phrasing for queries.

4. Describe Namespace

The sequence of the database and collection names is called a namespace.

5. What type of NoSQL database is MongoDB?

A document-oriented database is MongoDB. The information is kept in BSON structure-oriented databases. These documents are kept in a collection.

6. Describe MongoDB's indexes.

- To efficiently execute queries in MongoDB, we utilize indexes;
- In the absence of indexes, MongoDB should perform a collection scan, which involves scanning every document in a collection to identify the documents that match the query expression.
- MongoDB will use an index to limit the number of documents it should review if an appropriate index is available for a query.

**Take your Knowledge
Test Report**

Check your Score



7. Explain the importance of the covered query

Implementing a covered query speeds up the process since the indexes are kept in RAM or are arranged sequentially on the disk. It speeds up the execution of queries. Every field in the index is included in the covered query, and MongoDB returns the result fields after matching the query condition.

Check out: [MongoDB Course Syllabus](#)

8. Define replica set.

The replica can be defined as a collection of Mongo instances hosting comparable datasets. There will be two nodes in the replica set: a primary

and a secondary. From the primary to the secondary nodes, we duplicate every piece of data.

9. Describe the main and backup replica sets.

The nodes that allow writing are known as primary nodes in MongoDB. Master nodes are another name for primary nodes. In MongoDB, replication consists of a single master. As a result, only one node will be able to accept write operations simultaneously.

10. How do you handle transactions or locking in MongoDB?

Because MongoDB is fast, intuitive, and lightweight in its display, it does not employ conventional locking with the reduction.

It can be thought of as the MySQL auto-entrust script for MyISAM. We can improve the speed, especially in the structure with multiple servers, by using a simpler business model.

Check out: [MongoDB Salary in Chennai](#)

List of MongoDB Interview Questions for Experienced

11. What does the save() method accomplish?

12. Describe splitting.

13. How can we use MongoDB for sorting?

14. What is the skip() method's syntax?

15. What is the MongoDB ObjectID structure?

16. How can elements of an array with numerous criteria be found?
17. How is a nested join queried?
18. In the Nested Array, how do we update the object?
19. Explain how sharding works.
20. What is MongoDB's Aggregation Framework?
21. What does "Transactions" mean in MongoDB?
22. How can MongoDB databases be retrieved using a Javascript array?
23. What are Charts for MongoDB?
24. Which MongoDB backup and restoration tools are available?

MongoDB Technical Interview Questions and Answers for Experienced

11. What does the `save()` method accomplish?

We replace the current documents with new ones using the `save()` method.

12. Describe splitting.

The background procedure we employ to prevent chunks from being too big is called splitting.

13. How can we use MongoDB for sorting?

The *find()* method is used to locate any data in MongoDB. The collection of documents that we call this function over is returned using the *discovery ()* method.

To limit the output using MongoDB projection, we can utilize the “Where” clause in the MongoDB query.

```
db.<collection_name>.find({ }, {<key_Name>:<Flag to display>})
```

Check out: [MongoDB Tutorial for Beginners](#)

14. What is the skip() method's syntax?

The syntax for the skip() method in MongoDB is:

```
db.COLLECTION_NAME.find().limit(NUMBER).skip(NUMBER)
```

15. What is the MongoDB ObjectId structure?

The “_id” column in MongoDB is linked to ObjectId, and MongoDB sets this value as the default for the “_id” field in documents. To create an “ObjectId,” we employ the below syntax:

```
ObjectId([SomeHexadecimalValue])
```

Example:

```
ObjectId() = newObjectId
```

ObjectId provides the following methods:

Str: The object ID's string representation is provided in this method.

valueOf(): This function yields the ObjectId's hexadecimal representation.

getTimestamp(): This function yields the ObjectId's timestamp.

toString(): This function yields the ObjectId's string representation in "ObjectId(haxstring)".

**Take your Knowledge
Test Report**

Check your Score



16. How can elements of an array with numerous criteria be found?

For instance, assuming we possess the following documents:

```
{ _id: 1, numbers: [1000, -1000]}
```

```
{ _id: 2, numbers: [500]}
```

Upon executing the following command:

```
db.example.find( { numbers: { $elemMatch: { $gt: -10, $lt: 10 } } } );
```

17. How is a nested join queried?

We utilize "tested" to query the nested join.

Example:

```
{ "_id" : ObjectId( "abcd" ),
```

```
  "className" : "com.myUser",
```



```
“reg” : 12345,
```

```
“test” : [
```

```
{ “className” : “com.abc”,
```

```
“testid” : “pqrs” } ] }
```

18. In the Nested Array, how do we update the object?

The object is updated by running the following code:

Skip code block

```
{
```

```
“_id” : ObjectId(“4faaba123412d654fe83hg876”),
```

```
“user_id” : 123456,
```

```
“total” : 100,
```

```
“items” : [
```

```
{
```

```
“item_name” : “my_item_one”,
```

“price” : 20

},

{

“item_name” : “my_item_two”,

“price” : 50

},

{

“item_name” : “my_item_three”,

“price” : 30

}

]

}

19. Explain how sharding works.

- Data splitting between machines is known as sharding. This idea is also occasionally referred to as “partitioning.”
- By putting a portion of data on each machine, we can handle greater load and store more data without needing larger or more powerful machines.
- With MongoDB’s sharding feature, you may divide a collection among several machines (shards) by placing a subset of the data on each shard.
- This enables your application to expand beyond a solitary server’s or replica set’s resource constraints.

20. What is MongoDB’s Aggregation Framework?

You can do analytics on documents in one or more collections using the aggregation framework, a suite of analytics tools included with MongoDB.

The idea of a pipeline is the foundation of the aggregation framework.

- When using an aggregation pipeline, documents from a MongoDB collection are fed through one or more stages, each of which processes the inputs differently.
- Every step uses the output from the stage before it as its input.
- A continuous flow of documents serves as the input and output for every step.

21. What does “Transactions” mean in MongoDB?

A transaction is a logical processing unit in a database that consists of one or more read or write database operations. In MongoDB, transactions offer a helpful feature that helps to guarantee consistency.

Two APIs are available from MongoDB for transaction use.

Core API: The syntax is comparable to that of relational databases (start_transaction, commit_transaction, etc.).

Call-back API: The suggested method for utilizing transactions is this one. It initiates a transaction, carries out the designated actions, and then commits (or aborts on error).

Additionally, it automatically includes “TransientTransactionError” and “UnknownTransactionCommitResult” error-handling logic.

22. How can MongoDB databases be retrieved using a Javascript array?

To obtain the current databases, we can execute “Show DBS” on the MongoDB terminal. We run the following code to programmatically get the MongoDB databases:

```
use admin
```

```
dbs = db.runCommand({listDatabases: 1})
```

```
dbNames = []
```

```
for (var i in dbs.databases) { dbNames.push(dbs.databases[i].name) }
```

```
var connection = new Mongo();
```

```
var dbNames = connection.getDBNames();
```

23. What are Charts for MongoDB?

A brand-new, integrated data visualization tool in MongoDB is called MongoDB Charts.

The best method for utilizing data from a MongoDB database to produce visualizations is through MongoDB Charts.

It enables users to quickly represent data from a database without having to write code in a programming language like Python or Java.

The two distinct ways that MongoDB Charts are implemented are:

- MongoDB Charts PaaS (Platform as a Service)
- MongoDB Charts Server

24. Which MongoDB backup and restoration tools are available?

There are no backup, restore, export, or import features in the Mongo shell. Nevertheless, MongoDB has developed techniques to achieve this, negating the need for intricate GUIs or coding work.

To do this, many utility scripts are offered that may be used to import or export data in bulk from the database. These scripts for utilities are:

- mongoimport
- mongoexport
- mongodump
- mongorestore

Conclusion

You have reviewed these MongoDB Interview Questions and Answers. This revision helps before a technical interview. Recruiters often ask how MongoDB stores and manages application data. Questions may cover collections, queries, indexing, and document structure. Reading MongoDB

Interview Questions and Answers helps you recall these points during interview discussions.

Practice also improves understanding. Working with real databases makes the concepts clearer. If you want stronger database skills, you can join our [MongoDB Training in Chennai](#) and learn how to work with MongoDB through practical sessions.