

TOP 20 MongoDB INTERVIEW QUESTIONS 2022

Advertisements

Managing a database is becoming increasingly complex now due to the vast amount of data being produced continuously. Therefore a number of database management systems have been released. One of these is MongoDB which was released in 2009. Despite being relatively new in the field, it is doing extremely well. Usually, organizations don't prefer to switch their database management systems. But MongoDB made a lot of companies switch their systems due to its ease of usage. This is the reason why MongoDB developers are in great demand these days.

But to achieve your aim and become a MongoDB developer, you need to prepare from the very beginning. The competition is so fierce now that you can't succeed without making an edge over others. But with preparation and the right directions, you can beat that competition. And to help with your interview, we have prepared this comprehensive collection of MongoDB Interview Questions.

Top 20 MongoDB Interview Questions and Answers

What is MongoDB?

MongoDB is an open-source database management system for NoSQL. It is a cross-platform and document-oriented database. It is written in C++ language. It ditches the conventional table-based relational databases and uses documents like JSON instead with optional schemas. It's based on the concept of document and collection. Extreme scalability is provided by MongoDB through its features like sorting, range queries, indexes, aggregations, etc.

List the main characteristics of MongoDB.

Some of the main characteristics of MongoDB are given below-

- **Sharding**– Data can be split up across multiple machines by use of this feature.
- **Indexing**– Generic secondary indexes are supported by this feature. It provides various types of indexing such as geospatial, compound, unique, and full text.
- **File storage**– MongoDB facilitates easy storage of file metadata as well as large files.
- **Aggregation**– MongoDB provides an aggregation framework on the basis of data processing pipelines.
- **Special collection and index types**– Time-to-live data collections are supported by this feature for the kind of data that must be expired at a specific time.

What is meant by a document and a collection in MongoDB?

A document in MongoDB is a collection of keys along with their values arranged in an orderly manner. A map, directory, or hash is used to represent it. Objects represent the document in JavaScript, such as `{"greeting": "Hello World!"}`. Multiple keys and values are contained in a complex document, such as `{"greeting": "Hello World!", "views": 2}`.

A collection in MongoDB refers to a set of documents. Just like the document is analogous to a relational database's row, a collection is analogous to a table. They can have diverse shapes and schemas. An example is-

```
{"greeting": "Hello World!", "views": 2}
{"signoff": "Have a good day"}
```

What do you mean by Replica Set?

Replication is used for keeping identical copies of our data on multiple servers. This is helpful when something goes wrong with one or more servers. In such scenarios, the data remains safe. And the Replica set is used for such replication in MongoDB. A replica set contains a primary server and multiple secondary servers. In case of the primary server crash, a new primary is selected from the secondary servers themselves.

Differentiate between primary and secondary nodes in MongoDB?

Primary Nodes	Secondary nodes
We can write as well as read in these types of nodes, also called master nodes. Write operations can be accepted by only one node since it's a single replication master.	Secondary nodes are also referred to as slave nodes. This is because they allow only reading replicated from the primary nodes.

To learn more about MongoDB and its practices, [MongoDB Training](#) will help you to gain in-depth knowledge of the technology.

How does MongoDB scale-out occur?

Splitting up data across multiple servers is easy in MongoDB since it is a document-based model. MongoDB loads and balances the data across a cluster. Then the documents are redistributed automatically. An interface is provided by the query router of mongo between the sharded cluster and client applications. The metadata and settings are stored in the Config servers for clusters. There is a separate config server for each sharded cluster.

How can you perform a query in MongoDB?

A query can be performed using the **find** method in MongoDB. A subset of documents from a whole set of documents is returned by MongoDB. From no document to the whole collection, we can have anything we want. The first argument of **find** is the query criterion which is used to determine the document that is to be returned.

What is meant by Geospatial indexes in MongoDB?

There are two types of geospatial indexes in MongoDB- 2d and 2dsphere. Spherical geometries that model the earth's surface based on the WGS84 datum are used by the 2dsphere indexes. This works on the model of the earth as an oblate spheroid where the poles are flat. More accurate measures of distance are provided by 2dsphere than 2d indexes since it takes into account the earth's shape. 2d indexes are, on the other hand, used for storing points on a 2-dimensional plane.

What is meant by Transactions?

A transaction refers to the logical processing unit of a database capable of including multiple write operations or read operations. This ensures consistency in MongoDB. There are two types of APIs provided by MongoDB for using transactions-

- **Core API-** It has syntax like a relational database. **For example-** `start_transaction` and `commit_transaction`.
- **Call-back API-** This one is recommended for transactions. The transaction is started, executed, and committed by it. The error handling logic is automatically incorporated by it.

List the data types of MongoDB.

There is a wide array of data types supported by MongoDB. It includes several additional data types along with the keys and values of JSON. The main ones are as follows-

- a) Null- {"x": null}
- b) Number- {"x": 5}
- c) Boolean- {"x": false}
- d) Date- {"x": new Date()}
- e) String- {"x": "foobar"}
- f) Array- {"x": ["d", "e", "f"]}
- g) Binary data
- h) Embedded document- {"x": {"foo": "bar"}}

What are charts in MongoDB?

These are new tools for data visualization in MongoDB. It is the best visualization method. We don't even need to write any code in Python or Java to represent the database in the form of charts. MongoDB provides the following two types of implementations-

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

What is meant by a Mongo shell?

We can interact with MongoDB using this JavaScript shell through a command line. Administrative functions like exploring MongoDB or inspecting an instance can be performed using this shell. The following commands need to be run to start a shell-

```
$ mongod
$mongo
MongoDB shell version: 4.2.0
connecting to: test
>
```

The basic math works as follows-

```
>x=100;  
100  
>x/5;  
20
```

When should MongoDB be used?

MongoDB is used to build scalable business and internet applications. It should be chosen when a person wants-

- The data repository is to be scaled to a huge size.
- Rapid iterative development to be supported.
- The deployment type is to be evolved as there are changes in the business.
- The data is to be scaled to high levels of written and read traffic.
- The data is to be stored, managed, and searched with geospatial, text, and time-series dimensions.

What is meant by a SET modifier in MongoDB?

The value is set by “**\$set**” if there doesn’t exist a field value. This helps extremely to add keys defined by users and update schemas. **Example**–

```
>db.users.findOne()  
{  
  "_id": ObjectId("eafadf")  
  "name": "dinesh",  
  "age": 30,  
  "sex": "male",  
  "location": "India"  
}
```

Now field can be added as follows-

```
>db.users.updateOne({"_id": ObjectId("eafadf")},
```

```
...{"$set": {"favorite color": "Blue"}})
```

What utilities are available for backing up and restoring MongoDB?

Importing, exporting, restoring, and backing up are not included in the mongo shell functions. But there are methods defined by MongoDB for the purpose which doesn't need scripting complex GUIs. Hence, there are many utility scripts available to get bulk data in and out of the database. Some of these are mentioned below-

- mongoimport
- mongoexport
- mongorestore
- mongodump

What restrictions are there in the 32-bit version of MongoDB?

The storage size of the data and indexes contained server is 2 GB in MongoDB's 32-bit version. That's why MongoDB can't be deployed for production work on the machines with the 32-bit model. On the other hand, there is no restriction on the deployment of MongoDB's 64-bit version to the storage size. Hence, 64-bit operating systems are generally recommended.

When are the embedded data model and normalized data model, respectively, used in MongoDB?

The embedded data model is used when-

- a) There are “contains” relationships among the entities.
- b) There are one-to-many relationships among entities. The child or the “many” documents are displayed in relationships in the context of documents belonging to the parents.

And a normalized data model is used when-

- a) The duplication of data results from embedding, and sufficient read advantages are not there to take benefit.
- b) More tough many-to-many relationships are to be represented.
- c) The huge hierarchical data sets are to be modelled.

What is a capped collection in MongoDB?

It is a special type of collection in MongoDB. The collection size can be restricted in this type. Its syntax goes like-
`db.createCollection(<collection_name>, {capped: Boolean, autoIndexId: Boolean, max: Number, size: Number})`.

The given fields are there in the syntax of Capped Collection-

- **Collection_name**– it is the name of the capped collection that we make.
- **Capped**– It is a Boolean field, and it should be true if we create a capped collection. But it is false by default.
- **Size**– The maximum quantity of data is represented by this parameter in terms of bytes. It is an essential component of the capped collection.
- **Auto indexed**- This is again a Boolean flag. Indexes will be created when their value is set as true. On the other hand, indexes will not be created when their value is set as false.
- **Max**– The highest number of documents that are permitted inside the collection is represented by the Max parameter.

Differentiate between RDBMS and MongoDB.

Basis for comparison	MongoDB	RDBMS
Definition	This is a non-relational database management system.	This is a database system that is relational.
Working	It works with table relationships using rows and columns.	It is a document-based database system that works using documents and fields.
Performance	The performance is increased with the processors' rise.	The performance is increased with the RAM capacity rise.
Scalability	It is both vertically and horizontally scalable.	It is scalable only vertically.
Query Language	BSON is used by it to query the database.	SQL is used by it to query the database.
Hierarchical Data Storage	There is a built-in provision available for storing hierarchical data.	Storing hierarchical data is a bit tough here.
JavaScript Support	The JavaScript-based clients are supported by it to query the database.	The JavaScript-based clients are not supported by it to query the database.

Explain the different types of indexes that are supported by MongoDB.

There are many kinds of indexes supported by MongoDB. Some of them are listed below-

- **Default**– It is the “_id” created by MongoDB.
- **Multy-key**– the array data can be indexed by it.
- **Single-field**- The indexing and sorting take place over a single field by this type of index.
- **Compound**– Multiple fields can be indexed using this type of index.
- **Hashed**– The hashes can be indexed for multiple fields by using this index.
- **Geospatial**– The location data can be queried by using this kind of index.

Summary

MongoDB is a highly advanced database management system that is rich with features such as sharding and indexing. But that doesn't compromise its smooth functioning. Therefore it is in high demand these days in the industry. There is ample space for you to grow in this field. That's why make sure that you grab this opportunity this time only. These MongoDB interview Questions are for taking you one step closer to your dream job and company. Now, leave no stone unturned in its preparation and come out with flying colours from the interview process.