

100 MongoDB interview questions to hire top engineers

Questions

1. What is MongoDB, in simple terms?
2. How is MongoDB different from a regular spreadsheet or SQL database?
3. Can you describe what a document is in MongoDB?
4. What does it mean that MongoDB is a NoSQL database?
5. What is a collection in MongoDB? How does it relate to tables in SQL?
6. What are the advantages of using MongoDB?
7. How do you install MongoDB on your computer?
8. What is the MongoDB shell and how do you use it?
9. Explain how you would insert a new document into a MongoDB collection.
10. How do you find a specific document in MongoDB?
11. What is the purpose of the `_id` field in a MongoDB document?
12. How would you update an existing document in MongoDB?
13. What is the command to delete a document from MongoDB?
14. Can you give an example of a real-world scenario where MongoDB would be useful?
15. What are some basic data types supported by MongoDB?
16. What is indexing in MongoDB, and why is it important?
17. Explain what is meant by 'scalability' in the context of databases, and how MongoDB addresses it?
18. What are some tools you can use to interact with MongoDB besides the command line?
19. How can you back up and restore data in MongoDB?
20. What are some common mistakes beginners make when working with MongoDB, and how can you avoid them?
21. What is MongoDB?
22. Explain the difference between SQL and NoSQL databases, focusing on MongoDB.
23. What are documents and collections in MongoDB?
24. How would you insert a new document into a MongoDB collection?
25. How do you query all documents in a MongoDB collection?
26. How can you filter documents based on a specific field in MongoDB?
27. What is the purpose of the `_id` field in MongoDB documents?
28. How do you update a document in MongoDB?
29. How do you delete a document from a MongoDB collection?
30. Explain what indexing is and why it's important in MongoDB.
31. How do you create an index on a field in MongoDB?
32. What is the MongoDB shell and how do you use it?
33. Describe a situation where you might choose MongoDB over a relational database.
34. What are some common data types supported by MongoDB?
35. What is a MongoDB database?
36. What are some benefits of using MongoDB?
37. How do you connect to a MongoDB server from the command line?
38. What is a projection in MongoDB and how do you use it?
39. How do you sort the results of a query in MongoDB?
40. Explain the concept of embedded documents in MongoDB.
41. How do you use the `$exists` operator in MongoDB queries?
42. What is the purpose of the `$in` operator in MongoDB?
43. How do you use the `$and` and `$or` operators in MongoDB?
44. How can you limit the number of documents returned by a query?
45. What is the difference between `updateOne` and `updateMany` in MongoDB?
46. How do you use the `$set` operator in an update operation?
47. What is the purpose of the `upsert` option in the `updateOne` method?
48. Explain how you would use aggregation pipelines in MongoDB for simple data processing.
49. Describe a time when you faced a challenge while learning MongoDB, and how did you overcome it?
50. What are some tools you can use to visualize MongoDB data?
51. Explain the concept of sharding in MongoDB and when would you consider implementing it?
52. How does MongoDB handle transactions, and what are the different types of transactions supported?
53. Describe the role of indexes in MongoDB performance and how to identify inefficient queries needing indexing.
54. What are the differences between MongoDB and relational databases (like MySQL or PostgreSQL)?
55. How would you back up and restore a MongoDB database?
56. Explain the use of aggregation pipeline in MongoDB, providing an example of a complex data transformation.
57. What are the different write concerns available in MongoDB, and how do they affect data durability?
58. How does MongoDB ensure data consistency, and what are the trade-offs between consistency and performance?
59. Describe the use of MongoDB Compass and its advantages in managing and querying data.
60. Explain the purpose of the MongoDB profiler and how to use it to identify slow queries.
61. How would you design a schema for storing time-series data in MongoDB?
62. What are the different types of indexes in MongoDB, and when should you use each type?
63. How does MongoDB handle concurrency, and what are the different locking mechanisms used?
64. Explain the concept of replica sets in MongoDB and how they provide high availability.
65. How would you monitor the performance of a MongoDB cluster and identify potential bottlenecks?
66. What are the best practices for securing a MongoDB database, including authentication and authorization?
67. How does MongoDB handle large documents, and what are the limitations on document size?
68. Explain the use of TTL (Time-To-Live) indexes in MongoDB and how they can be used for data expiration.
69. How would you implement a full-text search in MongoDB?
70. What are the different data types supported by MongoDB, and how do they compare to other databases?
71. How would you optimize a MongoDB query that is performing poorly?
72. Explain the concept of the Oplog in MongoDB and its role in replication.
73. How would you handle schema evolution in a MongoDB database without causing downtime?
74. What are the considerations for choosing the right hardware for a MongoDB deployment?
75. How does MongoDB handle geospatial data and what are the different geospatial queries supported?
76. Explain the use of change streams in MongoDB and how they can be used for real-time data processing.
77. How would you migrate data from a relational database to MongoDB?
78. What are the differences between MongoDB Atlas and self-managed MongoDB deployments?
79. How would you implement data validation in MongoDB?
80. Explain the concept of GridFS in MongoDB and when would you use it?
81. How does MongoDB ensure data consistency across multiple replica set members, especially during network partitions?
82. Explain the concept of sharding in MongoDB. What are the key components involved, and how do they interact?
83. Describe your experience with MongoDB aggregation framework. Provide a specific example where you used it to solve a complex data processing problem.
84. What are the different types of indexes available in MongoDB, and how do you choose the right index for a given query?
85. How do you monitor the performance of a MongoDB deployment, and what metrics are most important to track?
86. Explain how you would design a MongoDB schema for a social media application with features like posts, comments, and likes. Focus on scalability and performance.
87. What are the trade-offs between using embedded documents and references in MongoDB, and when would you choose one over the other?
88. Describe a situation where you had to optimize a slow-performing MongoDB query. What steps did you take to identify and resolve the issue?
89. How does MongoDB handle concurrency, and what are the different types of locks used?
90. Explain your understanding of MongoDB's WiredTiger storage engine and its features like compression and encryption.
91. How would you implement a full-text search feature in MongoDB, and what are the limitations of MongoDB's built-in full-text search?
92. Describe your experience with MongoDB Atlas, and what are the advantages of using a managed MongoDB service?
93. Explain how you would implement data validation in MongoDB, and what are the different approaches available?
94. What are the different ways to back up and restore MongoDB data, and what are the considerations for each approach?
95. How would you design a disaster recovery plan for a MongoDB deployment?
96. Explain your understanding of MongoDB's change streams feature and how you can use it to build real-time applications.
97. How would you handle data migration in MongoDB, especially when dealing with large datasets or schema changes?
98. Describe your experience with MongoDB Compass, and how it can be used for data exploration and query analysis.
99. Explain how you can secure a MongoDB deployment, including authentication, authorization, and network security.