92 MySQL interview questions that you should ask to hire top engineers

Questions

1. Can you explain what a database is, like I'm five?

2. What is MySQL, and why do people use it?

3. Imagine your toys are in a box. How would MySQL help you organize them if you had a HUGE number of toys?

4. What's a table in MySQL, and what kind of stuff goes inside it?

5. If you wanted to find all the red toys in your toy box, how would MySQL help?

6. What does it mean to 'query' a database?

7. What's a primary key, and why is it important?

8. Can you give an example of a data type in MySQL?

9. What's the difference between VARCHAR and TEXT data types?

10. What does SQL stand for?

11. Name a few commands in SQL that are commonly used.

12. What's an index in MySQL, and how does it help?

13. How can you sort the results of a MySQL query?

14. What is the purpose of the WHERE clause in a MySQL query?

15. What are joins used for in MySQL, and can you provide a simple example?

16. Explain the difference between INNER JOIN and LEFT JOIN.

17. What is the use of GROUP BY clause?

18. How does the HAVING clause differ from the WHERE clause?

19. What is a subquery in MySQL?

20. Describe what NULL means in MySQL. Is it the same as zero or an empty string?

21. How would you optimize a slow-running query in MySQL, and what tools would you use?

22. Explain the difference between MyISAM and InnoDB storage engines, and when would you choose one over the other?

23. What are indexes in MySQL, how do they work, and what are the trade-offs of using them?

24. Describe the different types of joins in MySQL (INNER, LEFT, RIGHT, FULL), and provide examples of when each would be used.

25. What is normalization in database design, and why is it important? Explain different normalization forms (1NF, 2NF, 3NF).

26. How do you handle transactions in MySQL, and what are ACID properties?

27. Explain the purpose of stored procedures in MySQL, and what are the advantages of using them?

28. What are triggers in MySQL, and provide an example of how they can be used?

29. How can you implement replication in MySQL, and what are the benefits of doing so?

30. Explain how to backup and restore a MySQL database.

31. What are views in MySQL, and when would you use them?

32. How do you handle errors and exceptions in MySQL stored procedures?

33. Describe the process of tuning MySQL server performance.

34. What are the different isolation levels in MySQL, and how do they affect concurrency?

35. Explain how to use EXPLAIN to analyze a MySQL query.

36. How would you implement full-text search in MySQL?

37. What is the purpose of the ĠROUP BỲ clause in MySQL, and how does it work with aggregate functions?

38. How can you prevent SQL injection attacks in your MySQL applications?

39. Explain how to use partitioning in MySQL to improve performance and manage large tables.

40. Describe the process of migrating a MySQL database to a new server.

41. What are user-defined functions (UDFs) in MySQL, and how can you create them?

42. How does MySQL handle transactions, and what are the different isolation levels?

43. Explain the concept of 'locking' in MySQL, and describe the different types of locks.

44. What are the differences between MyISAM and InnoDB storage engines?

45. Describe the process of query optimization in MySQL.

46. How do you improve the performance of slow queries in MySQL?

47. What are the advantages of using stored procedures in MySQL?

48. Explain how to implement replication in MySQL and what are the different replication topologies?

49. What are the different types of indexes in MySQL, and how do they impact performance?

50. How does MySQL handle character sets and collations?

51. Explain the concept of partitioning in MySQL and its benefits.

52. How do you back up and restore a MySQL database?

53. What are the key differences between VARCHAR and TEXT data types, and when would you use each?

54. Describe the process of setting up and managing user permissions in MySQL.

55. How do you monitor the performance of a MySQL server?

56. Explain the concept of 'connection pooling' and its benefits in MySQL environments.

57. How do you handle database migrations in MySQL?

58. Describe the process of setting up and using SSL for MySQL connections.

59. What are the different types of MySQL errors, and how do you handle them in your application?

60. Explain the concept of 'ACID properties' in the context of MySQL transactions.

61. How does MySQL handle full-text search?

62. Describe the difference between clustered and non-clustered indexes and when you might use them.

63. Let's say your MySQL database is experiencing high CPU usage. What steps would you take to diagnose and resolve the issue?

64. How would you design a system for real-time analytics on frequently updated MySQL data without impacting transactional performance?

65. Explain the intricacies of using MySQL's Group Replication in a multi-datacenter setup, addressing potential issues like network latency and split-brain scenarios.

66. Describe your approach to optimizing a MySQL database experiencing slow query performance due to a high volume of concurrent read and write operations.

67. How can you implement a robust audit logging system in MySQL to track data modifications and access, while ensuring performance is not significantly degraded?

68. Discuss the trade-offs between using different isolation levels in MySQL and how they affect concurrency and data consistency in a high-transaction environment.

69. Explain how you would approach migrating a large, mission-critical MySQL database to a new hardware infrastructure with minimal downtime.

70. How do you handle and recover from a situation where a critical MySQL server suffers a complete hardware failure?

71. Describe the steps you would take to troubleshoot and resolve a deadlock situation in a MySQL database.

72. How would you implement and manage a data warehousing solution using MySQL, considering factors like data extraction, transformation, and loading (ETL) processes?

73. Explain the differences between various MySQL storage engines (InnoDB, MyISAM, etc.) and when you would choose one over another for specific use cases.

74. How do you ensure data security and compliance (e.g., GDPR, HIPAA) when storing sensitive information in a MySQL database?

75. Discuss your experience with MySQL performance tuning tools and techniques, such as query profiling, index optimization, and server configuration adjustments.

76. Describe how you would design a backup and recovery strategy for a MySQL database that meets specific recovery time objective (RTO) and recovery point objective (RPO) requirements.

77. How would you go about optimizing slow-running stored procedures in MySQL?

78. Explain how you can implement a custom caching layer on top of MySQL to improve read performance.

79. Describe your approach to monitoring and alerting on key MySQL metrics to proactively identify and address potential issues.

80. Explain how you would implement a sharded MySQL database architecture to handle massive data growth and high traffic loads.

81. Discuss the challenges and solutions associated with upgrading a MySQL database to a newer version with minimal disruption to applications.

82. How would you implement a system for detecting and preventing SQL injection attacks in a MySQL-backed application?

83. Explain your understanding of the MySQL optimizer and how it chooses execution plans for queries.

84. How do you debug performance issues related to table locking in MySQL?

85. Explain strategies to optimize full-text search capabilities in MySQL.

86. Describe your approach to capacity planning for a growing MySQL database, considering factors like storage, memory, and CPU.

87. How would you design a system for automatically detecting and correcting data corruption in a MySQL database?

88. Explain how you can leverage MySQL's spatial data types and functions for location-based applications.

89. Discuss the use of MySQL's JSON data type and its advantages and disadvantages compared to traditional relational data storage.

90. How would you go about optimizing a complex query involving multiple joins and subqueries in MySQL?

91. Describe the steps you would take to diagnose and resolve a situation where a MySQL database is experiencing high CPU utilization.