Time left 0:18:39

Question 1
Not yet answered
Marked out of 1.00

$lookup performs poorly when:

- ○ a. The foreign collection is sharded
- ○ b. The local field is unique
- [✓ ● c. Both collections share the same shard key]
- ○ d. The join is performed on indexed fields

Clear my choice

Question 2
Not yet answered
Marked out of 1.00

A MongoDB index with sparse: true will NOT index documents where:

- [✓ ● a. Index field does not exist]
- ○ b. Index field value is 0
- ○ c. Index field is null
- ○ d. Index field exists but is empty

Clear my choice

Question 3
Not yet answered
Marked out of 1.00

A query using LIKE '%abc' cannot use a B-Tree index because:

- ○ a. abc contains lowercase letters
- ○ b. The wildcard appears at the end
- ○ c. LIKE never uses indexes
- ○ d. The wildcard appears at the beginning

Question 4
Not yet answered
Marked out of 1.00

If _id index exists, which situation will cause a COLLSCAN even when filtering on _id?

- ○ a. Query uses _id: { $in: [1,2,3] }
- ○ b. Query uses $regex on _id
- ○ c. _id contains strings
- ○ d. Query uses _id: { $gte: 10 }

Question 5

Not yet answered

Marked out of 1.00

In a sharded MongoDB cluster, the config server primarily stores:

- ○ a. Write-ahead logs
- ○ b. All user data
- ○ c. Chunk metadata and cluster configuration
- ○ d. Index definitions

Question 6

Not yet answered

Marked out of 1.00

In MongoDB, when using $facet, which of the following is TRUE?

- ○ a. $facet can only be used on sharded clusters
- ○ b. All pipelines inside $facet run sequentially
- ○ c. $facet disables all indexes
- ○ d. $facet allows multiple pipelines to run on the same input in parallel

Question 7

Not yet answered

Marked out of 1.00

In MongoDB, which of the following queries can use a compound index on { age: 1, score: -1 } most efficiently?

- ○ a. db.users.find({ score: { $gte: 20 } }).sort({ score: 1 })
- ○ b. db.users.find({ age: { $gte: 20 } }).sort({ score: -1 })
- ○ c. db.users.find({ score: { $gt: 50 } }).sort({ age: 1 })
- ○ d. db.users.find({ age: { $gte: 20 } }).sort({ age: 1 })

Question 8

Not yet answered

Marked out of 1.00

In MongoDB, which situation makes a compound index { a: 1, b: 1 } unusable for sorting?

- ○ a. Query filters on a and sorts by { b: 1 }
- ○ b. Query filters on both a and b
- ○ c. Query sorts by { b: 1, a: 1 }
- ○ d. Query sorts by { a: -1, b: 1 }

Question 9
Not yet answered
Marked out of 1.00

In MySQL InnoDB, a secondary index lookup requires:

- ○ a. No lookup in clustered index
- ○ b. Reading only the secondary index
- ○ c. A single B-tree traversal
- ○ d. Reading both secondary index and clustered index

Question 10
Not yet answered
Marked out of 1.00

MongoDB multi-document ACID transactions require which storage engine?

- ○ a. RocksDB
- ○ b. MyISAM
- ○ c. WiredTiger
- ○ d. InnoDB

Question 11
Not yet answered
Marked out of 1.00

MySQL performs a full table scan when:

- ○ a. Partition key is present
- ○ b. Query references a non-indexed column in WHERE
- ○ c. Query returns fewer rows
- ○ d. Index fits entirely in memory

Question 12
Not yet answered
Marked out of 1.00

To force MySQL to use a specific index, you use:

- ○ a. INDEX FORCE
- ○ b. FORCE THIS INDEX
- ○ c. USE ONLY INDEX
- ○ d. USE INDEX

Question 13
Not yet answered
Marked out of 1.00

Which aggregation operator allows you to reshape documents by controlling inclusion/exclusion of fields?

- a. $reduce
- b. $map
- c. $merge
- d. $project

Question 14
Not yet answered
Marked out of 1.00

Which isolation level in MySQL prevents dirty reads but still allows non-repeatable reads and phantom reads?

- a. REPEATABLE READ
- b. SERIALIZABLE
- c. READ COMMITTED
- d. READ UNCOMMITTED

Question 15
Not yet answered
Marked out of 1.00

Which MongoDB feature ensures that writes go to the primary node before being replicated?

- a. Write Concern w:1
- b. Journaling
- c. Write Concern w:majority
- d. Read Preference primaryPreferred

Question 16
Not yet answered
Marked out of 1.00

Which MongoDB write concern guarantees that data is written to majority of replica set nodes before acknowledging?

- a. journaled
- b. w:1
- c. w:all
- d. w:majority

Question 17
Not yet answered
Marked out of 1.00

Which MySQL condition forces the optimizer to avoid using an index even if one exists?

- ○ a. column = 10
- ○ b. column IN (1,2,3)
- ○ c. WHERE column > 0
- ○ d. WHERE function(column) = value

Question 18
Not yet answered
Marked out of 1.00

Which MySQL join returns rows that have matching values in both tables but excludes unmatched rows?

- ○ a. INNER JOIN
- ○ b. FULL OUTER JOIN
- ○ c. RIGHT JOIN
- ○ d. LEFT JOIN

Question 19
Not yet answered
Marked out of 1.00

Which MySQL storage engine does not support foreign keys?

- ○ a. NDB
- ○ b. MEMORY
- ○ c. MyISAM
- ○ d. InnoDB

Question 20
Not yet answered
Marked out of 1.00

Which of the following best describes EXPLAIN in MySQL?

- ○ a. It rewrites queries automatically
- ○ b. It executes the query with maximum optimization
- ○ c. It updates statistics for the optimizer
- ○ d. It shows the execution plan without running the query

Question 21

Not yet answered

Marked out of 1.00

Which of the following causes index intersection to be used?

- a. Shard key is compound
- b. Compound index exists
- c. Query matches on two fields that each have separate single-field indexes
- d. Query uses $lookup

Question 22

Not yet answered

Marked out of 1.00

Which of the following is true for clustered indexes in InnoDB?

- a. They require manual configuration
- b. They store full row data in the index
- c. They store only pointers to rows
- d. They can be disabled

Question 23

Not yet answered

Marked out of 1.00

Which operator allows you to execute pipeline stages for each document inside an array?

- a. $unwind
- b. $each
- c. $map
- d. $pipeline

Question 24

Not yet answered

Marked out of 1.00

Which query will lock the selected rows and prevent other transactions from reading them in InnoDB?

- a. SELECT ... FROM ... LOCK
- b. SELECT ... FROM ...
- c. SELECT ... FOR UPDATE
- d. SELECT ... AS LOCKED

Question 25

Not yet answered

Marked out of 1.00

Which SQL query guarantees eliminating duplicates before ordering the output?

- ○ a. SELECT UNIQUE * FROM table ORDER BY col;
- ○ b. SELECT * FROM table ORDER BY col DISTINCT;
- ○ c. SELECT ORDER DISTINCT * FROM table;
- ○ d. SELECT DISTINCT * FROM table ORDER BY col;