emetack®
EXPLORE ENDLESS
POSSIBILITIES
TOGETHER

DATABASE PROGRAMMING

Instructions

- Complete all tasks below.

TOPIC: INDEXING AND PERFORMANCE TUNING

QUESTION

1. Creating an Index

- Write a SQL statement to create an index on the `last_name` column of the `employees` table to speed up searches.

2. Analyzing Query Performance

- Write a query and then use an `EXPLAIN` statement to analyze its execution plan. Discuss what the execution plan reveals about performance.

3. Removing Duplicate Indexes

- Given a scenario where there are duplicate indexes on the `orders` table, write a SQL statement to drop one of the duplicate indexes.

4. Using Composite Indexes

- Write a SQL statement to create a composite index on the `department_id` and `salary` columns of the `employees` table.

5. Indexing Strategy for a Table

- Explain the considerations you should take into account when deciding which columns to index in a table and provide an example.

6. Impact of Indexes on Data Modification

- Write a query and discuss how adding an index might affect the performance of `INSERT`, `UPDATE`, and `DELETE` operations on that table.

7. Query Optimizations

- Given a slow-running query, suggest optimizations that could improve its performance. What changes would you make?

8. Database Statistics

- Write a SQL statement to gather statistics on the `products` table, and explain how this information can be used for performance tuning.

9. Partitioning Tables

- Discuss the concept of table partitioning and write a SQL statement to partition a sales table by `year`.

10. Using Temporary Tables for Performance

- Explain how and when to use temporary tables to improve query performance. Provide an example SQL statement that creates a temporary table.

11. Full-Text Indexing

- Write a SQL statement to create a full-text index on a `products` table's `description` column to enhance search capabilities.

12. Monitoring Slow Queries

- Explain how to identify slow queries using the database's performance monitoring tools, and what actions can be taken to resolve performance issues.

13. Advanced Indexing Techniques

- Discuss the differences between clustered and non-clustered indexes and write an example SQL statement for each type.

14. Using Database Hints

- Write a query that uses a database hint to force the query optimizer to choose a specific execution plan.

15. Regular Maintenance for Indexes

- Explain the importance of index maintenance (e.g., rebuilding or reorganizing indexes) and provide a SQL statement to rebuild an index on the `sales` table.

END