1.ATTEMPT ALL QUESTIONS IN SECTION A / (55 Marks)

Answer all questions. Each question carries varying marks from 1 to 5.

1. Define the term "database server" and explain its role in a networked environment. (3 marks)

A database server is a specialized computer system that hosts and manages databases. It serves as a central repository for storing and retrieving data in a networked environment. Its role includes managing access to the database, executing queries, enforcing data integrity, and ensuring data security.

2. List and briefly describe three advantages of using a dedicated database server. (4 marks)

I. Performance: Dedicated database servers offer optimized performance as they are specifically designed to handle database operations efficiently.

ii. Scalability: With a dedicated database server, resources can be easily scaled up or down to accommodate changes in workload or data volume.

iii. **Security:** Dedicated database servers provide enhanced security features such as access controls, encryption, and authentication mechanisms, ensuring data integrity and confidentiality.

3. Explain the concept of database normalization and discuss its importance in database design. (5 marks)

Database normalization is the process of organizing data in a relational database to reduce redundancy and dependency. It involves breaking down large tables into smaller, related tables and defining relationships between them.

Importance: Database normalization improves data integrity, reduces data duplication, and enhances database performance by minimizing storage requirements and improving query efficiency.

4. Describe the process of installing and configuring a database server software. (5 marks)

The process of installing and configuring a database server software typically involves:

- Downloading the database server software from the official website.

- Running the installation wizard and selecting installation options (e.g., installation directory, components to install).

- Configuring server settings such as port number, authentication mode, and database storage location.

- Creating administrative user accounts and setting permissions.

- Testing the installation by connecting to the database server and executing basic queries.

5. What is the purpose of SQL injection attacks? Provide two methods to prevent SQL injection vulnerabilities in a database server. (4 marks)

SQL injection attacks aim to manipulate SQL queries executed by a database server to gain unauthorized access to data or execute malicious commands.

Prevention methods:

I. **Parameterized Queries:** Use parameterized queries or prepared statements to separate SQL code from user input, preventing attackers from injecting malicious SQL code.

ii. **Input Validation:** Validate and sanitize user input to ensure it conforms to expected formats and does not contain malicious SQL code.

6. Discuss the differences between relational and non-relational database servers. Provide examples of each. (5 marks)

Relational databases store data in structured tables with predefined schemas and support SQL for querying and manipulation (e.g., MySQL, PostgreSQL). Non-relational databases, also known as NoSQL databases, store data in flexible, schema-less formats and use various data models (e.g., document, key-value, graph) (e.g., MongoDB, Redis).

7. Explain the role of indexing in database performance optimization. Provide two examples of scenarios where indexing would be beneficial. (4 marks)

Indexing improves database performance by enabling faster data retrieval operations such as searching and sorting. Two scenarios where indexing would be beneficial:

I. **Search Queries**: Indexing columns frequently used in search queries (e.g., username, product name) improves query performance by reducing the time required to locate relevant records.

ii. **Join Operations:** Indexing foreign key columns used in join operations between related tables accelerates data retrieval by minimizing the need for full table scans.

8. Describe the steps involved in securing a database server against unauthorized access. (5 marks)

Steps for securing a database server against unauthorized access include:

- Implementing strong authentication mechanisms (e.g., using complex passwords, multi-factor authentication).

- Enforcing access controls to limit user privileges and restrict access to sensitive data.

- Encrypting data at rest and in transit to protect against unauthorized interception or access.

- Regularly applying security patches and updates to address vulnerabilities.

- Monitoring and logging database activity to detect and respond to security threats.

9. Discuss the concept of database backup and recovery. Why is it essential for a database server setup? (4 marks)

Database backup involves creating copies of database files or transactions to safeguard against data loss due to hardware failures, software errors, or human mistakes. Recovery involves restoring the database to a previous state using backup copies.

Essential for a database server setup:

Database backup and recovery are essential for ensuring data integrity, business continuity, and compliance with regulatory requirements. They provide a safety net against data loss and enable organizations to recover from unforeseen events quickly.

10. Compare and contrast the features of open-source and commercial database server software. (5 marks)

Open-source database server software (e.g., MySQL, PostgreSQL) is freely available and provides flexibility for customization and community support. Commercial database server software (e.g., Oracle, SQL Server) typically requires licensing fees and offers advanced features, support, and vendor-backed reliability.

11. What are the key considerations when choosing hardware for a database server setup? (3 marks)

Key considerations include:

- Processing power: Adequate CPU resources to handle database queries and transactions efficiently.

- Memory: Sufficient RAM to accommodate database caching and query execution.

- Storage: Fast and reliable storage subsystem for data persistence and retrieval.

- Network: High-speed network connectivity to support client-server communication.

- Redundancy: Redundant components (e.g., power supplies, storage) to minimize downtime.

12. Explain the term "data integrity" in the context of a database server. How can data integrity be maintained? (4 marks)

Data integrity refers to the accuracy, consistency, and reliability of data stored in a database. It ensures that data remains unchanged and reliable throughout its lifecycle.

Maintaining data integrity involves:

- Enforcing constraints (e.g., primary keys, foreign keys) to prevent invalid data entries.

- Implementing data validation rules to ensure data conforms to predefined standards.

- Performing regular data validation and verification checks to identify and correct inconsistencies.

- Utilizing transaction management and logging mechanisms to track and maintain data changes.

13. Discuss the role of transactions in database management systems. Provide an example of a transaction and explain its ACID properties. (5 marks)

Transactions in database management systems (DBMS) ensure data consistency and integrity by grouping database operations into atomic units of work. An example of a transaction is transferring funds from one bank account to another.

ACID properties of transactions:

- **Atomicity:** Transactions are atomic, meaning they are treated as a single unit of work that either succeeds or fails entirely.

- **Consistency:** Transactions maintain database consistency by preserving data integrity and adhering to predefined constraints.

- **Isolation**: Transactions are isolated from each other to prevent interference and ensure data integrity, even when multiple transactions are executed concurrently.

- **Durability:** Once a transaction is committed, its changes are permanently saved to the database, ensuring durability and data persistence.

14. Describe the process of database replication and explain its significance in a distributed database environment. (5 marks)

Database replication involves copying and maintaining identical copies of a database across multiple servers to enhance availability, reliability, and performance. It ensures data redundancy and fault tolerance, allowing for failover and load balancing in a distributed database environment.

15. Explain the concept of database clustering and discuss its advantages in terms of scalability and high availability. (5 marks)

Database clustering involves grouping multiple database servers together to function as a single logical unit. It enhances scalability and high availability by distributing workload across multiple nodes,

17. Discuss the role of database monitoring and performance tuning in maintaining optimal database server performance. (4 marks)

Database monitoring involves continuously monitoring database server metrics (e.g., CPU usage, memory usage, query performance) to identify bottlenecks and performance issues. Performance tuning involves optimizing database configurations, indexing strategies, and query execution plans to improve overall database performance.

2. ATTEMPT THREE QUESTIONS IN SECTION B / (30 Marks)

Answer any three questions. Each question carries 10 marks.

18. Discuss the steps involved in setting up user accounts and permissions in a database server environment.

Setting up user accounts and permissions involves:

i. Creating user accounts with unique usernames and passwords.

ii. Assigning appropriate privileges and roles to each user account.

iii. Defining access control lists (ACLs) to restrict access to specific databases, tables, or columns.

iv. Implementing role-based access control (RBAC) to manage permissions based on user roles and responsibilities.

v. Regularly reviewing and updating user accounts and permissions as needed.

19. Explain the process of database schema design and normalization with examples.

Database schema design involves organizing data into tables, columns, and relationships based on the requirements of the application. Normalization is the process of structuring the database schema to minimize redundancy and dependency.

Example:

Consider a library database with tables for books, authors, and borrowers. Normalization would involve breaking down the tables into smaller, related tables and

defining relationships between them (e.g., a separate table for authors with author ID and name).

20. Describe the role of stored procedures and triggers in enhancing database functionality.

Stored procedures are precompiled SQL statements stored in the database and executed on demand. They improve database performance by reducing network traffic and enhancing security.

Triggers are database objects that automatically execute in response to specified events (e.g., insert, update, delete) on a table. They enforce data integrity and implement complex business logic within the database.

3. ATTEMPT ONE QUESTION IN SECTION C / (15 Marks)

Answer one question. This question carries 15 marks.

21. Describe the process of implementing database mirroring for high availability and disaster recovery purposes. Discuss the benefits and limitations of database mirroring in a production environment.

Database mirroring involves creating and maintaining an identical copy (mirror) of a database on a separate server. It provides high availability and disaster recovery capabilities by automatically failing over to the mirror in case of a primary database failure.

Benefits:

Improved availability and uptime

Redundancy and fault tolerance

Disaster recovery capabilities

Limitations:

Requires additional hardware and network resources

Increased complexity and management overhead

Limited scalability compared to other high availability solutions

22. Discuss the role of data encryption in securing sensitive information stored in a database server. Explain the different types of encryption methods used in database servers and their implications for data security.

Data encryption protects sensitive information stored in a database server by converting it into ciphertext, which can only be decrypted with the appropriate decryption key.

Types of encryption methods:

Symmetric encryption: Uses a single encryption key for both encryption and decryption. Examples include AES and DES.

Asymmetric encryption: Uses a pair of public and private keys for encryption and decryption. Examples include RSA and ECC.

Hashing: Converts data into a fixed-length hash value that cannot be reversed. Examples include MD5 and SHA.

Implications for data security:

Symmetric encryption offers faster performance but requires secure key management.

Asymmetric encryption provides stronger security but is slower and computationally intensive.

Hashing is irreversible and used for verifying data integrity but does not provide encryption.