RDBMS Concepts

Concept-Based Questions

Q1. What is a Database Management System (DBMS)?
A1. A DBMS is software used to store, manage, and retrieve data in databases.

Q2. What is a Relational Database Management System (RDBMS)?
A2. An RDBMS is a type of DBMS based on the relational model which organizes data into tables (relations).

Q3. What are the properties of a relational database?
A3. Properties include data stored in tables, relationships maintained via keys, and adherence to normalization.

Q4. Define Primary Key and Foreign Key.
A4. Primary Key uniquely identifies each record. Foreign Key is a reference to a primary key in another table.

Q5. What is normalization and why is it important?
A5. Normalization is the process of organizing data to reduce redundancy and improve data integrity.

Q6. What are the different normal forms in database normalization?
A6. Common forms include:

- 1NF (Eliminate repeating groups)
- 2NF (Eliminate partial dependencies)
- 3NF (Eliminate transitive dependencies)
- BCNF (Stronger version of 3NF)
- Higher forms (4NF, 5NF) are rarely used

Q7. What is denormalization and when is it used?
A7. Denormalization is the process of combining normalized tables to improve read performance. It introduces redundancy for the sake of speed.

Q8. What are the types of data models in DBMS?
A8. The main types are:

- Hierarchical Model
- Network Model
- Relational Model
- Object-Oriented Model

Q9. What is an Entity-Relationship (ER) model?
A9. The ER model is a diagrammatic approach to database design using entities (things), attributes (properties), and relationships (associations).

Q10. What are cardinality and participation in ER modeling?
A10. Cardinality defines the number of instances of one entity related to another. Participation refers to whether all instances of an entity must be involved in the relationship (total or partial participation).

Code-Based Questions

Q11. Create a table ``** with student_id as Primary Key.**

```
CREATE TABLE students (
  student_id INT PRIMARY KEY,
  name VARCHAR(50),
  age INT
);
```

Q12. Create a table ** with a foreign key referencing **.**

```
CREATE TABLE courses (
  course_id INT PRIMARY KEY,
  student_id INT,
  course_name VARCHAR(50),
  FOREIGN KEY (student_id) REFERENCES students(student_id)
);
```

Q13. Insert data into the ``** table.**

```
INSERT INTO students (student_id, name, age)
VALUES (1, 'John', 21);
```

Q14. Write a query to fetch all students with their enrolled courses.

```
SELECT s.name, c.course_name
FROM students s
JOIN courses c ON s.student_id = c.student_id;
```

Q15. Add a UNIQUE constraint on the course_name column.

```
ALTER TABLE courses ADD CONSTRAINT uc_course UNIQUE (course_name);
```

Scenario-Based Questions

Q16. You notice duplicate entries in a table. What steps would you take?
A16. Check for missing constraints like PRIMARY or UNIQUE. Use GROUP BY or ROW_NUMBER() to detect and clean duplicates.

Q17. Explain how you would model a one-to-many relationship between authors and books.

A17. Create an authors table with author_id as PK. Create a books table with a foreign key referencing author_id.

Data Modeling and Database Design Concepts

Concept-Based Questions and Answers:

Q1. What is a Data Model?
A1. A data model defines how data is connected, stored, and accessed within a database system. It provides a conceptual framework for database design.

Q2. What are the main categories of data models?
A2. The three primary categories are: Conceptual Data Model, Logical Data Model, and Physical Data Model.

Q3. What is the role of a Conceptual Data Model?
A3. It focuses on identifying high-level relationships and entities without detailing the actual database structure.

Q4. What is a Logical Data Model?
A4. It outlines detailed data structure including attributes, primary and foreign keys, without implementation specifics.

Q5. What is a Physical Data Model?
A5. It represents how data is actually stored in the database including tables, indexes, partitions, and constraints.

Q6. What are the key stages of data modeling?
A6. The stages include: Requirement Analysis, Conceptual Modeling, Logical Modeling, Physical Modeling, and Implementation.

Q7. What does the Logical Data Model contain?
A7. It contains entities, attributes, relationships, normalization rules, keys, and data types.

Q8. What is the importance of using tools like Erwin?
A8. Erwin Data Modeler helps visually design, document, and manage data models and ensures consistency and accuracy.

Q9. What is the process of converting logical to physical model?
A9. It involves mapping entities to tables, attributes to columns, applying data types, constraints, and indexes based on performance needs.

Q10. Why is requirement analysis critical in data modeling?
A10. It ensures that the database design aligns with business needs and captures all necessary information accurately.

Q11. What is normalization in database design?
A11. It is the process of organizing data to reduce redundancy and improve data integrity by dividing tables based on functional dependency.

Q12. Explain First Normal Form (1NF).
A12. A table is in 1NF if it contains only atomic values and each column contains values of a single type.

Q13. Explain Second Normal Form (2NF).
A13. A table is in 2NF if it is in 1NF and all non-key attributes are fully functionally dependent on the primary key.

Q14. Explain Third Normal Form (3NF).
A14. A table is in 3NF if it is in 2NF and has no transitive dependency between non-prime attributes.

Q15. What is denormalization?
A15. It is the process of combining normalized tables to improve read performance by introducing some redundancy.

Q16. When is denormalization preferred?
A16. In scenarios where performance and quick data retrieval are more important than update efficiency.

Q17. What is Specialization in data modeling?
A17. It is the process of defining a set of sub-entities from a single higher-level entity (supertype).

Q18. What is Generalization in data modeling?
A18. It is the process of combining common attributes of multiple entities into a single higher-level entity.

Q19. What is an Entity in ER modeling?
A19. An entity represents a real-world object or concept, typically mapped to a table in SQL Server.

Q20. What is an Attribute in ER modeling?
A20. Attributes define the properties or details of an entity and correspond to columns in a SQL Server table.

Introduction to ANSI SQL and MySQL

Concept-Based Questions

Q1. What is ANSI SQL and how is it different from MySQL?
A1. ANSI SQL is a standard language for relational database management systems. MySQL is a relational database management system that implements SQL with its own specific features and extensions.

Q2. What are the key features of MySQL?
A2. Key features include open-source nature, support for multiple storage engines, high performance, replication, strong security, and compatibility with many platforms.

Q3. Why is ANSI SQL important?
A3. ANSI SQL ensures portability and standardization across different RDBMS platforms.

Q4. What are DDL, DML, and DCL commands?
A4.

- DDL: Data Definition Language (CREATE, ALTER, DROP)
- DML: Data Manipulation Language (INSERT, UPDATE, DELETE)
- DCL: Data Control Language (GRANT, REVOKE)

Q5. Explain the difference between SQL and PL/SQL.
A5. SQL is a query language to interact with databases. PL/SQL is Oracle's procedural extension to SQL that includes control structures.

Code-Based Questions

Q6. Write a SQL statement to create a database named ``.

```
CREATE DATABASE company_db;
```

Q7. How do you display all databases in MySQL?

```
SHOW DATABASES;
```

Q8. Write a SQL query to use a specific database.

```
USE company_db;
```

Q9. Display the current date in MySQL.

```
SELECT CURDATE();
```

Q10. Show the version of MySQL currently in use.

```
SELECT VERSION();
```

Scenario-Based

Q11. A developer is unsure which SQL syntax will work across all RDBMS. What do you recommend?

A11. Recommend using ANSI-standard SQL syntax and avoiding vendor-specific extensions.