Unit-1: Introduction to Fundamentals of DBMS

1. **What are some common database applications?**
    - Common database applications include banking systems, airline reservation systems, telecommunications databases, financial systems, sales and marketing databases, and online retail systems.
2. **What is the primary purpose of database systems?**
    - The primary purpose of database systems is to store, manage, and retrieve data efficiently and securely while ensuring data integrity and providing a centralized location for data access.
3. **Can you describe the main components of a DBMS?**
    - The main components of a DBMS include the database engine, database schema, query processor, transaction management, storage management, and database access language.
4. **Explain the DBMS architecture.**
    - DBMS architecture typically includes three levels: internal (physical storage), conceptual (logical structure), and external (user views). It can be centralized, distributed, or client-server based.
5. **What are different data models in DBMS?**
    - Different data models include the hierarchical model, network model, relational model, object-oriented model, and document model.
6. **Define data independence and its types.**
    - Data independence refers to the ability to change the schema at one level without affecting the schema at the next higher level. It has two types: logical data independence (changing conceptual schema without affecting external schema) and physical data independence (changing internal schema without affecting conceptual schema).

Unit-2: Database Design and ER Model

1. **What are the key steps in the database design process?**
    - The key steps include requirements analysis, conceptual design (ER modeling), logical design (schema definition), and physical design (storage and indexing).

2. **Explain the entity-relationship model.**
    - The entity-relationship model (ER model) is a conceptual framework that describes the structure of a database using entities, attributes, and relationships.
3. **What constraints can be applied in an ER model?**
    - Constraints in an ER model include key constraints, participation constraints, domain constraints, and referential integrity constraints.
4. **How are ER diagrams useful in database design?**
    - ER diagrams visually represent the entities, attributes, and relationships in a database, helping in the organization and planning of database structure.
5. **What are weak entity sets?**
    - Weak entity sets are entities that cannot be uniquely identified by their own attributes alone and depend on a "strong" or "owner" entity for identification.
6. **Describe the extended features of ER models.**
    - Extended features include generalization, specialization, aggregation, and categorization, which allow for more complex data modeling.

Unit-3: Relational Databases

1. **What is the relational model?**
    - The relational model organizes data into tables (relations) of rows and columns, where each table represents an entity and each row represents a record.
2. **Describe the structure of relational databases.**
    - Relational databases consist of tables with rows and columns, where each row represents a record and each column represents an attribute.
3. **What are the fundamental relational algebra operations?**
    - Fundamental relational algebra operations include selection, projection, union, set difference, Cartesian product, and renaming.
4. **How do additional and extended relational algebra operations differ?**
    - Additional operations include intersection and natural join, while extended operations include division and various join operations like outer join.
5. **Explain the concept and use of views in relational databases.**

- Views are virtual tables that are defined by a query and provide a way to present data without storing it physically. They can simplify complex queries and provide security by restricting access to specific data.

6. **What are DDL and DML statements in SQL?**
    - DDL (Data Definition Language) statements define the structure of the database, such as CREATE, ALTER, and DROP. DML (Data Manipulation Language) statements manage data within the database, such as SELECT, INSERT, UPDATE, and DELETE.

Unit-4: SQL (DDL)

1. **What is the purpose of Data Definition Language in SQL?**
    - The purpose of DDL is to define, alter, and manage the structure of database objects like tables, indexes, and schemas.
2. **List some common data types used in SQL.**
    - Common data types include INT, VARCHAR, CHAR, DATE, TIMESTAMP, FLOAT, and BOOLEAN.
3. **How do you define a schema in SQL?**
    - A schema in SQL is defined using the CREATE SCHEMA statement, which groups database objects under a specific namespace.
4. **Describe the basic structure of SQL queries.**
    - The basic structure of SQL queries includes the SELECT clause (to specify columns), FROM clause (to specify tables), WHERE clause (to filter rows), and other optional clauses like GROUP BY, HAVING, and ORDER BY.
5. **Explain the functions of CREATE, ALTER, and DROP commands in SQL.**
    - CREATE is used to create new database objects, ALTER modifies existing objects, and DROP removes objects from the database.
6. **What are the uses of RENAME and TRUNCATE commands in SQL?**
    - RENAME is used to change the name of a database object, while TRUNCATE removes all rows from a table quickly without logging individual row deletions.

Unit-5: SQL (DML)

1. **What are the main DML commands in SQL?**

- The main DML commands are SELECT, INSERT, UPDATE, and DELETE.

2. **How do you perform SELECT operations in SQL?**
    - SELECT operations are performed by specifying the columns to retrieve, the table to query from, and optional filtering conditions.
3. **Describe how to INSERT data into a database.**
    - Data is inserted using the INSERT INTO statement, specifying the target table and the values for each column.
4. **How do you DELETE and UPDATE records in SQL?**
    - DELETE removes records based on specified conditions, while UPDATE modifies existing records based on specified conditions.
5. **Explain how constraints are implemented in SQL.**
    - Constraints are implemented using keywords like PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, and CHECK within table definitions.
6. **What are nested subqueries and how are they used?**
    - Nested subqueries are queries within another query, used to perform operations like filtering and aggregation based on results from another query.

Unit-6: Relational Languages

1. **What is Tuple Relational Calculus?**
    - Tuple Relational Calculus (TRC) is a non-procedural query language that specifies what to retrieve based on a specified condition.
2. **Describe Domain Relational Calculus.**
    - Domain Relational Calculus (DRC) is a non-procedural query language that specifies queries using domain variables, focusing on the attributes rather than tuples.
3. **How does Query by Example (QBE) work?**
    - QBE allows users to create queries by filling out a template or example form, making it easier to construct queries visually.
4. **Explain the concept of Data log.**
    - Data log is a declarative logic programming language used for deductive databases, combining logic programming with database capabilities.
5. **What are the set operations in relational databases?**

- Set operations include UNION, INTERSECT, and EXCEPT, which combine results from multiple queries.

6. **How are aggregate functions used in SQL?**
    - Aggregate functions like COUNT, SUM, AVG, MAX, and MIN are used to perform calculations on sets of values, typically used with GROUP BY.

Unit-7: Relational Database Design

1. **What are the key features of relational database design?**
    - Key features include normalization, data integrity, minimized redundancy, and efficient query performance.
2. **Explain atomic domains and the first normal form.**
    - Atomic domains ensure that each attribute contains indivisible values. First normal form (1NF) requires that all attribute values are atomic.
3. **Describe the theory of functional dependency.**
    - Functional dependency describes a relationship where one attribute uniquely determines another attribute within a relation.
4. **How is decomposition using functional dependencies performed?**
    - Decomposition involves breaking down a relation into smaller relations to eliminate redundancy and maintain dependencies, ensuring lossless join and dependency preservation.
5. **What are multivalued dependencies?**
    - Multivalued dependencies occur when one attribute in a relation determines a set of values for another attribute independently of other attributes.
6. **What are the steps involved in the database design process?**
    - Steps include requirements analysis, conceptual design, logical design, schema refinement, physical design, and implementation.

Unit-8: Transaction Management

1. **Define the concept of a transaction in DBMS.**
    - A transaction is a sequence of operations performed as a single logical unit of work, ensuring data consistency and integrity.
2. **What are the different states of a transaction?**

- Transaction states include active, partially committed, committed, aborted, and failed.

3. **How are atomicity and durability implemented in DBMS?**
    - Atomicity ensures that all operations within a transaction are completed successfully, while durability ensures that committed transactions are permanently recorded.
4. **Explain concurrent execution in DBMS.**
    - Concurrent execution allows multiple transactions to execute simultaneously, improving system performance and resource utilization.
5. **What is serializability in the context of transactions?**
    - Serializability ensures that the outcome of concurrent transactions is the same as if they were executed sequentially in some order.
6. **How is recoverability ensured in DBMS?**
    - Recoverability ensures that a transaction can be undone or redone to maintain database consistency in case of failures.

Unit-9: Concurrency Control

1. **What are lock-based protocols?**
    - Lock-based protocols use locks to control access to data, ensuring that only one transaction can access data at a time to prevent conflicts.
2. **Describe timestamp-based protocols.**
    - Timestamp-based protocols assign timestamps to transactions to order them and ensure serializability by resolving conflicts based on these timestamps.
3. **What are validation-based protocols?**
    - Validation-based protocols perform checks at transaction commit time to ensure that transactions do not violate serializability, allowing optimistic concurrency control.
4. **How is deadlock handled in DBMS?**
    - Deadlock handling techniques include deadlock prevention, deadlock detection, and deadlock resolution, such as rolling back one or more transactions.
5. **What are weak levels of consistency?**

- Weak levels of consistency, like read uncommitted, read committed, repeatable read, and snapshot isolation, allow for trade-offs between performance and consistency.

6. **Explain insert and delete operations in the context of concurrency control.**
    - Insert and delete operations must be managed carefully to avoid conflicts and ensure that changes are visible to other transactions only when appropriate.

Unit-10: SQL (DCL/TCL)

1. **What is the purpose of the GRANT and REVOKE commands in SQL?**
    - GRANT is used to give users specific privileges on database objects, while REVOKE removes those privileges.
2. **How do ROLLBACK and COMMIT commands function?**
    - COMMIT saves all changes made during the transaction, making them permanent, while ROLLBACK undoes all changes made during the transaction.
3. **What is the use of SAVEPOINT in SQL?**
    - SAVEPOINT creates a point within a transaction to which you can roll back, allowing partial rollbacks without affecting the entire transaction.
4. **Describe the implementation of aggregate functions in SQL.**
    - Aggregate functions perform calculations on a set of values, such as COUNT, SUM, AVG, MAX, and MIN, typically used with GROUP BY clauses.
5. **How are inbuilt character functions used in SQL?**
    - Inbuilt character functions, such as UPPER, LOWER, SUBSTRING, and LENGTH, perform operations on string data types.
6. **Explain the usage of inbuilt numeric and date & time functions in SQL.**
    - Numeric functions, such as ABS, CEIL, FLOOR, and ROUND, perform operations on numeric data types. Date and time functions, such as NOW, DATEADD, DATEDIFF, and DATEPART, manipulate and retrieve information from date and time data types.

Unit-11: Recovery System

1. **What are the different types of failures in a DBMS?**

- Types of failures include transaction failures, system crashes, media failures, and application errors.

2. **Describe the storage structure in DBMS recovery.**
    - The storage structure includes volatile storage (main memory) and non-volatile storage (disk storage), with data usually stored in log files for recovery purposes.
3. **How is log-based recovery implemented?**
    - Log-based recovery uses logs to record changes made by transactions, allowing the system to redo or undo changes in case of failures.
4. **What challenges are faced with recovery and concurrent transactions?**
    - Challenges include ensuring consistency, managing interleaved operations, and maintaining isolation during recovery.
5. **Explain the role of buffer management in DBMS recovery.**
    - Buffer management handles the caching of data pages in memory, ensuring that modified pages are written to disk and managing the buffer pool during recovery.
6. **How is recovery managed with the loss of non-volatile storage?**
    - Recovery with the loss of non-volatile storage involves restoring data from backups and applying logged changes to bring the database to a consistent state.

Unit-12: Distributed Databases

1. **What is a distributed database?**
    - A distributed database is a collection of interconnected databases spread across multiple locations, appearing as a single database to users.
2. **How is data fragmentation used in distributed databases?**
    - Data fragmentation divides a database into smaller, logical pieces, called fragments, which are distributed across different sites for improved performance and availability.
3. **Explain data replication and allocation techniques.**
    - Data replication involves storing copies of data at multiple sites to ensure availability and reliability, while allocation determines the placement of data fragments across sites.

4. **What is a semi join in distributed databases?**
    - A semi join reduces the amount of data transferred between sites by only sending relevant data required to complete a join operation.
5. **What are the differences between homogeneous and heterogeneous databases?**
    - Homogeneous databases use the same DBMS at all sites, while heterogeneous databases use different DBMSs, requiring additional software for integration and communication.
6. **How are distributed transactions managed?**
    - Distributed transactions are managed using protocols like two-phase commit (2PC) to ensure atomicity, consistency, isolation, and durability (ACID properties) across multiple sites.

Unit-13: Cloud-Based Databases

1. **How did computing evolve from collaborative to cloud computing?**
    - Computing evolved from collaborative systems, which allowed multiple users to work together, to cloud computing, which provides scalable and on-demand resources over the internet.
2. **Describe client-server computing.**
    - Client-server computing involves clients requesting services and resources from centralized servers, which process and respond to these requests.
3. **What is peer-to-peer computing?**
    - Peer-to-peer computing is a decentralized network model where each participant (peer) can act as both a client and a server, sharing resources directly without centralized coordination.
4. **How does distributed computing differ from grid computing?**
    - Distributed computing involves multiple interconnected computers working together to solve tasks, while grid computing involves a network of computers working on large-scale problems, often using idle resources.
5. **Explain the concept of collaborative computing.**
    - Collaborative computing involves multiple users working together on shared tasks and resources, often facilitated by software tools that enable communication and collaboration.

6. **What are the key differences between distributed computing and cloud computing?**
    - Distributed computing focuses on distributing tasks across multiple computers, while cloud computing provides scalable and on-demand resources via the internet, often abstracting the underlying infrastructure from users.

Unit-14: Introduction to PL/SQL

1. **What is a PL/SQL block?**
    - A PL/SQL block is a logical unit of code in PL/SQL, consisting of a declarative part, an executable part, and an exception-handling part.
2. **How are conditional statements used in PL/SQL?**
    - Conditional statements in PL/SQL, such as IF-THEN-ELSE, allow for decision-making based on specified conditions.
3. **Describe the use of loops in PL/SQL.**
    - Loops in PL/SQL, such as FOR, WHILE, and LOOP-END LOOP, enable repeated execution of a block of code until a specified condition is met.
4. **What are cursors in PL/SQL and how are they used?**
    - Cursors in PL/SQL are used to handle query results one row at a time, allowing for row-by-row processing of result sets.
5. **Explain the role of triggers in PL/SQL.**
    - Triggers are stored procedures that automatically execute in response to certain events on a table or view, such as INSERT, UPDATE, or DELETE operations.
6. **How does PL/SQL enhance the capabilities of SQL?**
    - PL/SQL enhances SQL by adding procedural programming constructs, such as loops, conditions, and exception handling, allowing for more complex and flexible database operations.

4o