Chapter 1: Database Basics

1. What is Database Management System (DBMS)? Write the essential applications of DBMS.

2. Explain the difference between two-tier and three-tier database architectures. Which is better suited for web applications? Why?

3. Define and explain with examples: Super Key, Candidate Key, Primary Key, Foreign Key.

4. List five responsibilities of a database-management system manager and describe the problems if these responsibilities are not fulfilled.

5. What do you mean by RDBMS? Why would you not call traditional file systems a database? List the drawbacks of using file systems to store data.

6. What do you mean by Database Instances and Schemas?

7. What do you mean by Data Models? Describe different categories of data models.

Chapter 2: Database Design

1. Entity-Relationship Modeling

- What is the main purpose of an ER diagram? Explain the differences between weak and strong entity sets.
- Design a database for an airline system with an ER diagram and relational schema, including constraints.
- What is mapping cardinality? Explain with examples.

2. Normalization

- What are the steps involved in database normalization?
- Normalize a given relation (e.g., video rental database) to 1NF, 2NF, and 3NF.
- Given a relation R(A, B, C, D) and Functional Dependency set FD = {AB -> CD, B -> C}, determine whether R is in 2NF. If not, convert it into 2NF.

3. Constraints

- Define integrity constraints and referential integrity. Provide examples.

- What do you mean by self-referencing tables? Explain the distinction between total and partial constraints.

Chapter 3: Relational Algebra and SQL

1. Relational Algebra

- Write relational algebra for queries such as:
  - Find students of the "CSTE" department and their registered courses.
  - Find students who did not register for any courses.
  - Delete the record of a student with a specific student ID.
- Explain the difference between Cartesian Product and Natural Join Notation.

2. SQL Queries

Based on different schemas provided in the document, write SQL queries for the following:

- List all students of a specific department (e.g., CSTE).
- List all student names with their membership numbers.
- List details of students who borrowed a book whose author is "Humayun Ahmed."
- Count how many books have been borrowed by each student.
- List books taken by a student with a specific student ID.
- Find all customers of a bank who have an account but no loan.
- Find names of all customers who live on the same street and in the same city as "Smith."
- Find names of all branches with customers who have accounts in the bank and live in "Harrison."
- Find total people owning cars involved in accidents in a specific year (e.g., 2009).
- Add a new accident record to a database with specific attributes.
- Delete the Mazda car belonging to "John Smith."
- Select items whose price is between 500 and 1000.

- Find the maximum price of an item.

- Find the total quantity sold of a specific item (e.g., item ID = 7).

- Count how many machines have been allocated to the "CSIT" class.

- Retrieve machine allotment details of a student with a specific ID (e.g., student ID = 5).

- Count the machines allocated in a specific lab for a specific day (e.g., "Monday").

- How many students, class-wise, have been allocated machines in labs?

- Perform SQL-based operations on an insurance database schema to find relationships, such as accidents or participants.

3. Complex SQL Operations

- Display grades for students based on a marks table with score ranges (F, C, B, A).

- Count the number of students for each grade in the marks table.

- Construct a B+-tree step-by-step for a given dataset using SQL.

Chapter 4: Indexing and Storage

1. Why is indexing needed? Differentiate Dense Index and Sparse Index with examples.

2. Construct a B+ tree for a given dataset step-by-step.

3. Explain Hash Index and ideal properties of a hash function.

4. What are the differences between volatile, non-volatile, and stable storage types? How would you implement stable storage?

Chapter 5: Transactions and Deadlocks

1. Define transactions and explain the ACID properties.

2. Explain the concept of deadlocks, their detection, and approaches to prevent them.

3. Illustrate and explain a transaction state diagram.

4. What do you mean by conflict equivalence and conflict serializability? Use precedence graphs to determine serializability.

Chapter 6: Advanced Database Topics

1. What is Data Mining and Data Warehousing? Explain:

- Fact tables
- Dimension tables
- Star schema
- Snowflake schema
- Constellation schema

2. Discuss the benefits of concurrent execution and differentiate between serial and serializable schedules.

3. What is RAID? Explain the different RAID levels.

Chapter 7: Miscellaneous Topics

1. What are triggers in SQL? Why is it called the event-condition-action model?

2. Define and explain locking protocols and their phases (e.g., Growing Phase, Shrinking Phase).

3. Explain multi-valued and derived attributes with examples.