CMR
INSTITUTE OF
TECHNOLOGY

USN | | | | | | | | | | |

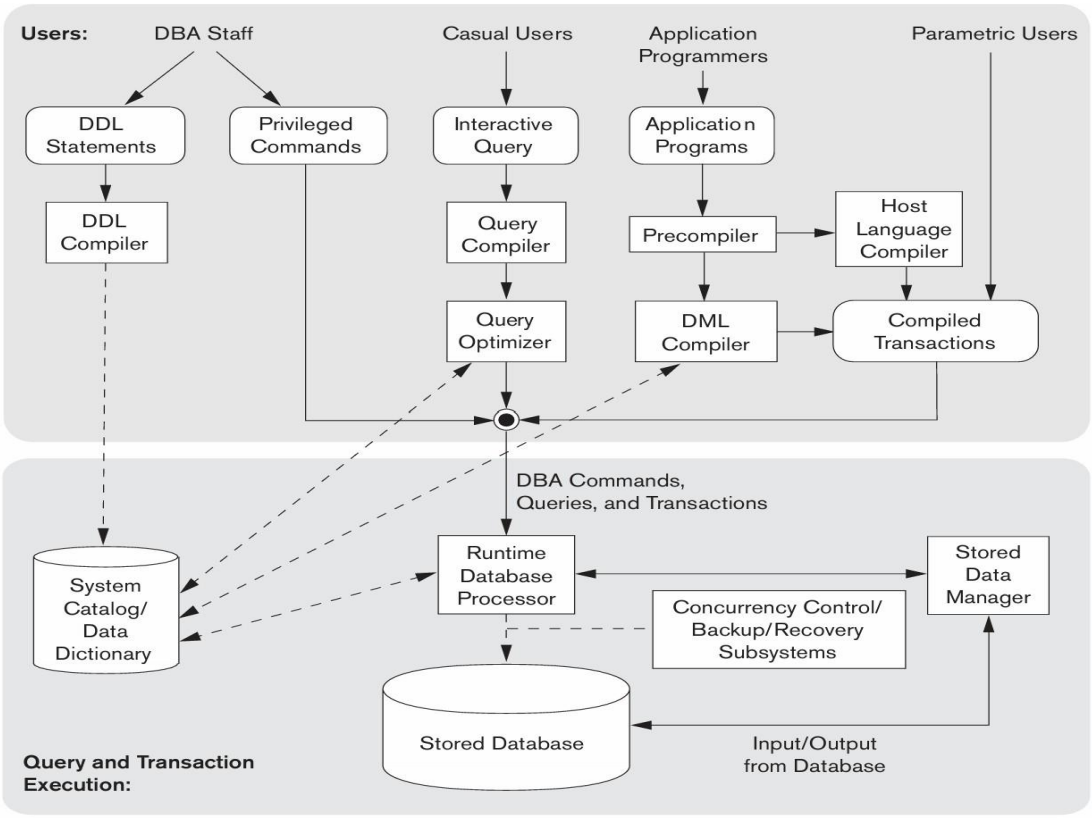
Internal Assessment Test I – Dec 2025

Sub:	Database Management Systems - Solution	Sub Code:	MMC103

Note: Answer FIVE FULL Questions, choosing ONE full question from each Module

		MARKS	OBE	
			CO	RBT
	PART I			
1	Explain the components of a DBMS and their functions. OR	[10]	CO1	L1
2	Examine participation constraints in the ER model. Differentiate between total and partial participation.	[10]	CO1	L3
	PART II			
3	Define the Entity–Relationship (ER) Model. Explain its components. OR	[10]	CO2	L2
4	What are views of data? Explain the external, conceptual, and internal views with examples.	[10]	CO2	L2
	PART III			
5	Draw an ER diagram for a **Library Management System** showing: a. BOOK, MEMBER, ISSUE b. Weak entity if any c. Relationship attributes OR	[10]	CO2	L3
6	Describe the set operations in relational algebra.	[10]	CO2	L2
	PART IV			
7	Differentiate between the ER Model and Relational Model. OR	[10]	CO2	L2
8	Discuss the different types of keys in the relational model with examples	[10]	CO2	L2
	PART V			
9	Explain the SELECT and PROJECT operations in relational algebra with detailed examples. OR	[10]	CO1	L1
10	Demonstrate the structure of SQL and classify SQL commands with examples.	[10]	CO2	L3

Q1. Explain the components of a DBMS and their functions.



1. DDL Statements
Used by DBA staff
Commands like CREATE, ALTER, DROP
Define the structure of the database
2. DDL Compiler
Converts DDL statements into internal form
Stores metadata into the System Catalog / Data Dictionary
3. Privileged Commands
Used only by DBA
Include:
Grant/revoke permissions
Backup and recovery commands
Performance tuning commands
4. System Catalog / Data Dictionary
Central repository of metadata
Stores:
Table names
Attribute types
Constraints
User permissions
Consulted during query processing and execution
5. Interactive Query
SQL queries issued by casual users
Example:
SELECT * FROM STUDENT;
6. Query Compiler
Parses SQL queries
Checks syntax and semantics
Converts query into an internal representation
7. Query Optimizer

Selects the most efficient execution plan
Decides:
Which index to use
Order of operations
Improves performance
8. Application Programs
Written by application programmers
Contain embedded SQL statements
Example: Java + SQL
9. Precompiler
Extracts SQL from application programs
Separates SQL and host language code
10. DML Compiler
Compiles DML commands (SELECT, INSERT, UPDATE, DELETE)
Converts them into low-level database instructions
11. Host Language Compiler
Compiles the remaining application code (Java, C, etc.)
Produces executable programs
12. Compiled Transactions
Precompiled and stored queries
Used by parametric users
Example: ATM withdrawal, ticket booking
13. Runtime Database Processor
Executes:
Queries
Transactions
DBA commands
Acts as the core execution engine
14. Concurrency Control / Backup / Recovery Subsystems
Ensure:
Concurrency control (multiple users at same time)
Transaction atomicity
Recovery after failure
15. Stored Data Manager
Manages actual data storage
Handles:
File access
Indexes
Disk I/O operations
16. Stored Database
Actual data stored on disk
Includes tables, indexes, and records
17. Input / Output from Database
Data transferred between disk and memory
Controlled by Stored Data Manager

Q2. Explain participation constraints in the ER model.

Participation constraint specifies whether all or some entity instances participate in a relationship.

Types of Participation

1. Total Participation

- Every entity **must participate** in the relationship.
- Represented using **double lines** in ER diagram.

Example:
Every EMPLOYEE must work in a DEPARTMENT.

2. Partial Participation

- Some entities **may not participate** in the relationship.
- Represented using **single line**.

Example:
Not all STUDENTs enroll in HOSTEL.

Difference Table

Aspect	Total Participation	Partial Participation
Mandatory	Yes	No
ER Notation	Double line	Single line
Example	Employee–Department	Student–Hostel

Q3. Define ER Model and explain its components.
The ER Model is a conceptual data model used to represent real-world objects and their relationships.

Components of ER Model

1. Entity
    - Real-world object (Student, Employee).
    - Represented by a rectangle.
2. Entity Set
    - Collection of similar entities.
3. Attributes
    - Properties of entities.
    - Types:
        - Simple
        - Composite
        - Derived
        - Multivalued
4. Key Attribute
    - Uniquely identifies an entity.
    - Example: Roll No.
5. Relationship

- Association between entities.
- Represented by diamond.

6. Relationship Set
    - Collection of similar relationships.
7. Cardinality
    - One-to-One
    - One-to-Many
    - Many-to-Many

Q4. Views of Data – External, Conceptual, Internal.

Views of data describe how data is perceived at different abstraction levels.

Three-Schema Architecture

1. External View

- User-specific view.
- Shows required data only.

Example:
Student sees only marks and attendance.

2. Conceptual View

- Logical structure of the entire database.
- Includes entities, attributes, and relationships.

Example:
Tables: STUDENT, COURSE, ENROLLMENT.

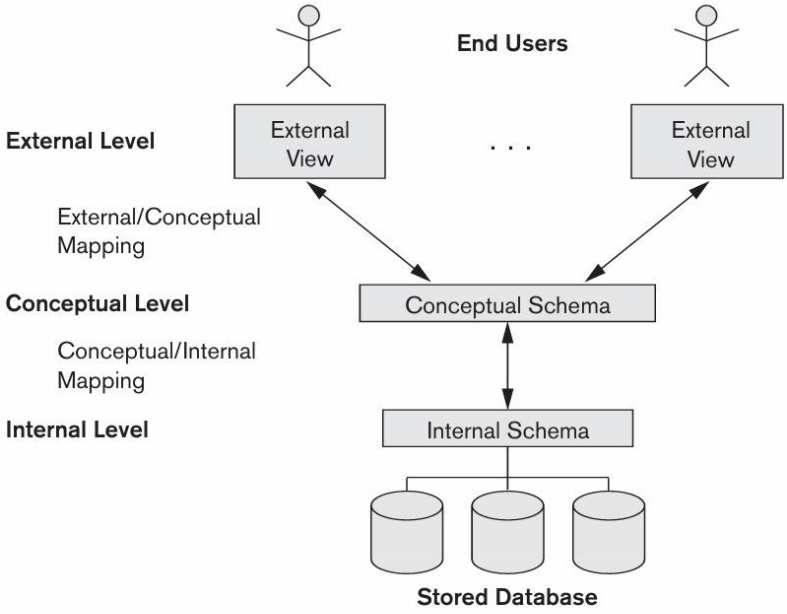
3. Internal View

- Physical storage of data.
- Includes indexing, file organization.

Example:
Data stored in B-trees on disk.

Advantages

- Data abstraction
- Security
- Data independence



Q5. ER Diagram – Library Management System.

1. BOOK (Strong Entity)

- Book_ID *(Primary Key)*
- Title
- Author
- Publisher
- Price

2. MEMBER (Strong Entity)

- Member_ID *(Primary Key)*
- Name
- Address
- Phone_No

3. ISSUE (Weak Entity)

- Issue_Date
- Return_Date
- Fine_Amount

ISSUE is a weak entity because it cannot exist without BOOK and MEMBER.
Diagram

Q6. Set Operations in Relational Algebra.
Set Operations

1. Union (∪)

- Combines tuples from two relations.
- Relations must be union-compatible.

2. Intersection (∩)

- Common tuples from both relations.

3. Difference (−)

- Tuples in first relation but not in second.

4. Cartesian Product (×)

- Combines every tuple of one relation with another.

Q7. ER Model vs Relational Model.

The ER Model is a high-level conceptual data model used to represent real-world entities and their relationships in a graphical form. It helps database designers understand system requirements clearly before database implementation.

Key Features

- Represents data using entities, attributes, and relationships
- Uses ER diagrams (rectangles, diamonds, ovals)
- Focuses on what data is required, not how it is stored
- Used during the database design phase

2. Relational Model

The Relational Model is a logical data model where data is stored in the form of tables (relations) consisting of rows and columns. It is based on mathematical concepts such as set theory and predicate logic.

Key Features

- Represents data using tables
- Uses rows (tuples) and columns (attributes)
- Relationships are implemented using foreign keys
- Supports SQL for data manipulation
- Used during database implementation

Aspect	**ER Model**	**Relational Model**
Level of Abstraction	High-level (Conceptual)	Logical/Low-level
Representation	ER diagram	Tables (Relations)

Aspect	ER Model	Relational Model
Basic Components	Entity, Attribute, Relationship	Table, Row, Column
Relationships	Shown using diamonds	Implemented using foreign keys
Keys	Key attributes	Primary key, Foreign key
Focus	Database design	Database implementation
Complexity	Easy to understand	More formal and strict
Used By	Database designers	DBMS and programmers
Supports SQL	No	Yes
Purpose	Requirement analysis	Data storage and retrieval

Q8. Types of Keys.
A key is an attribute or a set of attributes that is used to uniquely identify tuples (rows) in a relation (table). Keys help in maintaining data integrity and relationships between tables.

Types of Keys

1. Super Key

- A set of one or more attributes that uniquely identifies a tuple.
- May contain extra attributes.

Example:
STUDENT(Roll_No, Email, Name)

- {Roll_No}
- {Email}
- {Roll_No, Name}

2. Candidate Key

- Minimal super key.
- No attribute can be removed without losing uniqueness.

Example:

- Roll_No
- Email

3. Primary Key

- One candidate key selected to uniquely identify records.
- Cannot be NULL or duplicate.

Example:
STUDENT(Roll_No, Name, Dept)

4. Alternate Key

- Candidate keys not chosen as primary key.

Example:
If Roll_No is primary key, Email is alternate key.

5. Composite Key

- A key formed using two or more attributes.

Example:
ENROLLMENT(Roll_No, Course_ID)
→ (Roll_No, Course_ID)

6. Foreign Key

- Attribute that references the primary key of another table.
- Used to maintain referential integrity.

Example:
STUDENT(Roll_No)
MARKS(Roll_No) → Foreign Key

Q9. SELECT and PROJECT operations.

1. SELECT Operation (σ)

Definition

- Used to select rows (tuples) from a relation that satisfy a given condition.
- It is a horizontal operation.

Syntax

σ condition (Relation)

Example

EMPLOYEE(Eid, Name, Salary, Dept)

σ Salary > 30000 (EMPLOYEE)

➞ Returns employees whose salary is greater than 30000.

2. PROJECT Operation (π)

Definition

- Used to select specific columns (attributes) from a relation.
- It is a vertical operation.
- Removes duplicate tuples.

Syntax

π attribute_list (Relation)

Example

π Name, Salary (EMPLOYEE)

➞ Returns only Name and Salary columns.

Difference between SELECT and PROJECT

SELECT	PROJECT
Selects rows	Selects columns
Horizontal operation	Vertical operation
Uses condition	Uses attribute list
Symbol σ	Symbol π

Combined Example

π Name (σ Dept = 'MCA' (STUDENT))

Q10. Structure of SQL and classification.

SQL (Structured Query Language) is used to interact with relational databases.

```
SELECT column_list
FROM table_name
WHERE condition
GROUP BY column
HAVING condition
ORDER BY column;
```

- SELECT → columns to display
- FROM → table name

 WHERE → condition

 GROUP BY → grouping

 HAVING → group condition

 ORDER BY → sorting

Classification of SQL Commands

1. DDL – Data Definition Language

Used to define database structure.

Command	Example
CREATE	CREATE TABLE STUDENT(...);
ALTER	ALTER TABLE STUDENT ADD Age INT;
DROP	DROP TABLE STUDENT;

2. DML – Data Manipulation Language

Used to manipulate data.

Command	Example
INSERT	INSERT INTO STUDENT VALUES(...);
UPDATE	UPDATE STUDENT SET Dept='MCA';
DELETE	DELETE FROM STUDENT WHERE Roll_No=1;

3. DQL – Data Query Language

Used to retrieve data.

Command	Example
SELECT	SELECT * FROM STUDENT;

4. DCL – Data Control Language

Used for authorization.

Command	Example
GRANT	GRANT SELECT ON STUDENT TO user1;
REVOKE	REVOKE SELECT FROM user1;

5. TCL – Transaction Control Language

Used to manage transactions.

Command	Example
COMMIT	COMMIT;
ROLLBACK	ROLLBACK;
SAVEPOINT	SAVEPOINT sp1;