LESSON PLAN 2025-26(SUMMER)

NAME OF THE TEACHER : ANJALI MUNDA, GUEST LECT.(CSE)

Subject: **DATABASE MANAGEMENT SYSTEMS (Course Code: CSEPC 204/TH2)**
Program: Diploma in Computer Science and Engineering
Semester: 4th
Total Contact Hours: 45
Total Marks: 100
Assessment: Internal Assessment – 30, End Term – 70

After completion of the course, the students will be able to:

CO1-Explain the basic concept about the DBMS

CO2- databases using entity-relationship (ER) models, normalization techniques, and execute SQL queries for data manipulation and retrieval.

CO3-Analyze relational algebra to optimize database queries and ensure consistency, isolation, and durability through transaction management.

CO4- Evaluate database recovery techniques and security measures to ensure data integrity, performance, and protection from unauthorized access.

Lesson No.	UNIT	Topic/Sub-Topic	Learning Objective	Activity	Homework	COURSE OBJECTIVE
		UNIT-1-:Database Fundamentals (6 Hours)				
1	I	Definition & Need for DBMS	Define DBMS and identify why we moved away from manual records	Brainstorming: List data sets used in daily life (Spotify, Banking).	Find 3 examples of "hidden" databases you use daily.	**CO1**
2	I	DBMS vs. File System	Compare traditional file processing with modern DBMS.Differentiate between structured and unstructured storage.	T-Chart Creation: Map file system flaws (Redundancy, Isolation) to DBMS solutions.	Write a paragraph on the risks of storing bank data in Excel.	C01
3	I	MS Architecture & Components	Explain how the DBMS engine, metadata, and users interact.	Component Sketching: Diagramming the flow from a User Query to the Physical Disk	Research the specific role of the "Query Optimizer."	C01
4	I	Views of Data & 3-Schema	Distinguish between Physical, Logical, and View levels.	The "Abstraction" Game: Explain a database to a CEO vs. a Developer.	Draw the 3-schema architecture for a Hospital System.	CO1

5	I	Data Models (Hierarchical, Network, Relational)	Compare different ways to structure data relationships.	Relationship Mapping: Draw the same data as a Tree (Hierarchical) vs. a Table.	Write one major drawback of the Network model.	CO1
6	I	Intro to RDBMS: Pros & Cons	Define Relational DBMS and evaluate its efficiency.	SWOT Analysis: Group discussion on the cost vs. benefit of RDBMS.I	Install an open-source RDBMS (MySQL/PostgreSQL).	CO1
UNIT-2:- Introduction to ER Models: (8 hours)						
7	II	Entities & Attributes	Identify "Nouns" as entities and their properties as attributes. Model real-world objects into data entities.	Noun Hunting: Highlight entities and attributes in a real-world Netflix case study.	List 5 attributes for a "Vehicle" entity.	CO2
8	II	Relationships & Constraints	Define how entities interact (Verbs) and set cardinality Define data interactions and business rules..	String Game: Physically connect students (Entities) with yarn based on relationships.	Identify the cardinality (1:1, 1:N, M:N) for a Library system.	CO2
9	II	Super, Candidate, & Primary Key	Distinguish between various unique identifiers.	Key Sorting: Given a list of attributes (Name, SSN, Email), find the most "minimal" key.	Find a Candidate Key for a "Hospital Patient" table.	CO2
10	II	Foreign Keys & Referential Integrity	Understand how tables link to each other.	Linkage Lab: Create a "Customer" and "Order" table and link them via IDs	Explain what happens if a Primary Key is deleted while a Foreign Key exists.	CO2
11	II	ER Diagram Components	Master symbols: Rectangles, Ovals, Diamonds, and Lines.	Drafting: Draw a mini-ERD for a "Coffee Shop" on a whiteboard or paper.	Draw an ERD for a "Social Media Profile" (User, Post, Like).	CO2
12	II	Complex ERDs: Weak Entities & Participation	Model entities that depend on others (e.g., Dependents of Employees).	The Shadow Entity: Diagramming a "Hotel Room" that can't exist without a "Hotel."	Research the "double rectangle" symbol in ERDs.	CO2

13	II	Converting ERD to Relational Model (Part 1)	Transform Strong Entities and simple attributes into tables.	Table Translation: Convert a "Student" ERD into a SQL-ready table schema.	Map a "Product" entity with composite attributes to a table.	CO2
14	II	Converting Relationships to Tables (Part 2)	Map M:N relationships into "Junction/Bridge" tables.	The Bridge Build: Create a table that connects "Students" to "Courses."	Complete a full mapping of a "Hospital Management" ERD.	CO2
UNIT-3:-Relational Algebra & SQL :(9 hours)						
15	III	Relational Algebra: Select & Project	Master unary operations (σ and π) for filtering rows and columns.	Paper Filter: Given a paper table, physically "mask" columns (Project) and rows (Select).	Write the algebraic expression for: "Get names of students with GPA > 3.5."	CO3
16	III	Relational Algebra: Set & Product	Understand Union ($\cup$), Intersection ($\cap$), and Cartesian Product ($\times$).	Set Logic: Use Venn diagrams to visualize Union vs. Intersection of two tables.	Draw a Cartesian Product for a table with 2 rows and a table with 3 rows.	CO3
17	III	DDL: Create, Alter, Drop	Define and modify database structures using SQL.	Blueprint Design: Write SQL scripts to create a "Store" database schema.	Create a "Library" table with appropriate data types.	CO3
18	III	DML: Insert, Update, Delete	Manipulate data entries within existing tables.	Data Entry Lab: Populate the "Store" database and perform "Price Update" scenarios.	Write a query to delete all records where status = 'inactive'.	CO3
19	III	Filtering & Sorting (WHERE, ORDER BY)	Refine search results using logical operators and sorting.	The Search Engine: Query a "Movies" dataset to find titles from 2024 sorted by rating.	List the difference between BETWEEN and IN operators	CO3

20	III	Refine search results using logical operators and sorting.	Summarize data using SUM, AVG, COUNT, and GROUP BY.	Sales Report: Calculate total revenue per category from a raw sales list.	Find the average salary per department for a "Staff" table.	CO3
UNIT 4:- Database Design and Normalization: (8 hours)						
21	IV	Data Redundancy & Anomalies	Identify Update, Insertion, and Deletion anomalies.	The "Messy Spreadsheet": Students try to delete a record and accidentally lose unrelated info.	List 3 problems caused by duplicate data in a DB.	CO4
22	IV	Functional Dependencies (FDs)	Define $X \to Y$ and identify dependencies in a schema.	Dependency Mapping: Link attributes (e.g., SSN $\to$ Name) on a whiteboard.	Identify FDs in a "Library Book" dataset.	CO4
23	IV	First Normal Form (1NF)	Eliminate repeating groups and ensure atomicity.	Cell Splitting: Convert a table with multiple phone numbers in one cell into 1NF.	Rewrite a "Project Team" list into 1NF format.	CO4
24	IV	Second Normal Form (2NF)	Eliminate "Partial Dependencies" (Composite Keys).	Key Check: Identify attributes that only depend on part of a primary key.	Find a 2NF violation in a "Student-Course" table.	CO4
25	IV	Third Normal Form (3NF)	Eliminate "Transitive Dependencies."	Chain Breaking: Fix a table where $A \to B$ and $B \to C$ (e.g., EmpID $\to$ Dept $\to$ DeptHead).	Convert a 2NF table to 3NF.	CO4
26	IV	Boyce-Codd Normal Form (BCNF)	Handle advanced anomalies where determinants are not keys.	The Overlap Lab: Resolve complex dependencies in multi-key tables.	Research: When is 3NF better than BCNF?	CO4
27	IV	Decomposition of Relations	Break one large table into smaller ones without losing data.	Lossless Join Test: Try to "re-join" two tables to see if the original data is preserved.	Explain "Lossy Decomposition" in your own words.	CO4

28	IV	Full Normalization Case Study	Apply 1NF through BCNF to a real-world messy dataset.	The Big Fix: Groups compete to normalize a "Hospital Management"	Complete a final normalization report on the case study.	CO4
UNIT-5 - Database Transaction Management and Concurrency Control::(8 hours)						
29	V	Introduction & ACID Properties	Explain Atomicity, Consistency, Isolation, and Durability.	Banking Simulation: Walk through a fund transfer; what happens if the power goes out mid-way?	Write a real-world example of a "Consistency" violation.	CO5
30	V	Transaction States	Identify the lifecycle of a transaction (Active, Partially Committed, etc.).	State Mapping: Draw the transition diagram and identify the "Point of No Return."	Define the difference between 'Committing' and 'Aborting'.	CO5
31	V	Schedules & Serializability	Understand how multiple transactions interleave (Serial vs. Parallel).	Conflict Check: Analyze two transaction logs to see if they overlap safely.	Research: What is a "Conflict Serializable" schedule?	CO5
32	V	Recoverability	Differentiate between Recoverable, Cascadeless, and Strict schedules.	The Domino Effect: Simulate a "Cascading Rollback" where one failure kills 5 transactions.	Explain why "Cascadeless" schedules are preferred.	CO5
33	V	Lock-Based Protocols	Master Shared (S) and Exclusive (X) locks and 2-Phase Locking (2PL).	Locking Lab: Act as a "Lock Manager" for two students trying to edit the same file.	List the rules of the "Growing Phase" in 2PL.	CO5
35	V	Deadlocks	Identify, prevent, and detect Deadlock cycles.	The Traffic Jam: Draw a Wait-for-Graph (WFG) and find the cycle causing the "freeze."	Contrast Deadlock Prevention vs. Deadlock Detection.	CO5
36	V	Timestamp Ordering	Use system clocks/logic to order transactions without locks.	Timeline Sorting: Reorder transactions based on their "Entry Time" to resolve conflicts.	Research the "Thomas Write Rule."	CO5

Signature of Teacher   Signature of HOD

37	V	Recovery & Concurrency Wrap-up	Synthesize all concepts into a full system failure/recovery scenario.	The Crash Test: Given a log file and a system crash, determine which transactions to Redo or Undo.	Final Review: Compare Locking vs. Timestamping pros/cons.	CO5

UNIT-6 Database Administration and Security Management: (6 Hours)

38	VI	User Management in DBMS	Create and manage user accounts; understand the principle of least privilege	Live Demo: Create a new user in SQL and attempt to login with restricted access.	Research the difference between "SysAdmin" and "DBA" roles.	CO6
39	VI	Privileges and Roles	Distinguish between System and Object privileges; use Roles to simplify management.	Role-Play: Assign students "Manager" or "Clerk" roles and grant permissions to specific tables.	Write a script to create a "Read-Only" role and assign it to a test user.	CO6
40	VI	Backup Strategies	Identify types of backups (Full, Incremental, Differential) and their importance.	Scenario Analysis: Given a specific data loss event, choose the most efficient recovery path.	Create a weekly backup schedule for a small e-commerce database.	CO6
41	VI	Database Recovery	Understand the WAL (Write-Ahead Logging) and Point-in-Time Recovery (PITR).	Simulation: Walkthrough of a "Transaction Rollback" after a simulated system crash.	Define the terms RTO (Recovery Time Objective) and RPO (Recovery Point Objective).	CO6
42	VI	Authentication & Authorization	Compare Multi-Factor Authentication (MFA) vs. Biometrics; understand the AuthN/AuthZ flow.	Flowcharting: Map the path from login request to data access.	Find a recent news article about a database breach caused by poor authentication.	CO6
43	VI	Data Encryption	Implement TDE (Transparent Data Encryption) and understand Data Masking.	Cipher Lab: Encrypt a string of text using a simple AES-256 tool or manual SQL functions.	Explain the difference between "Data at Rest" and "Data in Transit" encryption.	CO6

Signature of Teacher **Signature of HOD**