Course code	Course Name	Hours/week			Credit	Max. Marks
		L	T	P	C	100
71203002003	**Database Management Systems**	3	0	1	4	

Prerequisite	Basic programming skills

Evaluation Scheme	**Theory & Practical**		**Hours**	**Marks**
	End Semester Examination		2	50
	Internal 1) Midterm – 10 Marks 2) CIA (Assignment/Certification/Quiz)-15 Marks 3) Attendance -5 Marks 4) Practical - 20 Marks			50

UNIT-I	**INTRODUCTION TO DBMS**	**4**

Introduction: An overview of database management system, Advantages and Applications, Database system vs Conventional File system, Database system Architecture, 3 Levels of Abstraction, Data models, Schema and Instance.

UNIT-II	**ER MODELLING AND NORMALISATION FORMS**	**12**

Data Modelling using Entity Relationship Model: ER model concepts, Notations for ER diagram, Mapping cardinalities, Keys and Constraints, Concepts of Super Key, Candidate key, Primary key and foreign key, Extended ER model, Generalization & Specialization, Aggregation, Reduction of ER diagrams to tables. Basics of SQL - DDL, and DML commands.
Functional dependencies, Armstrong's axioms, Normalization and its benefits, Normal forms, First Normal Form, Second Normal Form, Third Normal Form, BCNF, Fourth Normal Form, Fifth Normal Form, Dependency preserving decompositions, Lossless join decompositions.

UNIT-III	**STRUCTURED QUERY LANGUAGE - SQL AND PL/SQL**	**12**

Table creation, alteration, defining constraints – Primary key, foreign key, unique, not null, check, Functions, Aggregate functions, Built-in functions: Numeric, Date, String functions, set operations, Sub-queries, Nested queries, Use of group by and order by, Joins and its types. Views in SQL, Transaction control commands: Commit, Rollback, Savepoint. PL/SQL Concepts: Cursors, PL/SQL Data types, Variables, Literals, Constants, Stored Procedures, Functions, Triggers.

UNIT-IV	**TRANSACTIONS, CONCURRENCY CONTROL AND DEADLOCKS**	**11**

Transactions: ACID Properties, State Diagram, Serializability of schedules in database transactions, Consistent and Inconsistent state, Recoverable and Non-Recoverable Schedules, Cascading Rollback and Cascadeless Schedule, Recovery from transaction failures, Log based recovery, Checkpoints. Implementation of DCL and TCL commands.
Concurrency control: Lock based techniques, Shared and Exclusive Locks, Two-Phase locking Protocol, Role of concurrency control manager, lock manager.
Deadlocks: Deadlock Detection using wait-for-graph, Deadlock Recovery Schemes.

UNIT-V	**DATABASE SECURITY AND NoSQL DATABASES**	**6**

Database Security: Overview, Discretionary access control, Mandatory Access Control, Data Encryption. NoSQL Databases: Overview, history and applications of NoSQL Databases, Advantages of NoSQL, Distributed Systems, Benefits: Data Availability, Fault tolerance and Load balancing, BASE Properties, Four Types of NoSQL Databases, Difference between Relational and Non-Relational Databases, Column-Family Data Store Features. Database Project development.

	Total hours	**45** periods

Course Outcomes: At the end of the course, the students will be able to:

COs	Statements	Bloom's Level

CO1	Understand Database System Architecture concepts and the underlying concepts of its technologies, installation and instance setup containing a sample database.	**L2**
CO2	Create ER Diagrams as per problem statement, implement normal forms for designing a database and utilize DML & DDL commands.	**L3**
CO3	Solve any given practical database problem by implementing DML commands of SQL and PL/SQL including subqueries, joins, views, stored procedures and triggers.	**L3**
CO4	Analyze transactions processing, control concurrency techniques and locking protocols and implement the DCL & TCL commands.	**L4**
CO5	Evaluate Database Security, NoSQL Database technologies and create a working database application for a suitable use case by applying the project-based learning.	**L6**

TEXT BOOK:

1.	Silberschatz, A., Korth, H. F., & Sudarshan, S. (2002). *Database system concepts* (4th ed.). McGraw Hill.
2.	Elmasri, R., & Navathe, S. B. (2003). *Fundamentals of database systems* (3rd ed.). Pearson Education.
3.	Ramakrishnan, R., & Gehrke, J. (2003). *Database management systems* (3rd ed.). McGraw-Hill.

REFERENCES:

4.	Rob, P., & Coronel, C. (2003). *Database systems: Design, implementation, and management* (5th ed.). Thomson Learning, Course Technology.
5.	Gruber, M. (2008). *Understanding SQL* (1st ed.). BPB Publications.
6.	Koch, G., & Loney, K. (2002). *Oracle 9i – The complete reference* (Indian ed.). McGraw-Hill India.

ONLINE REFRENCES:

1.	Oracle Academy: Database Design, Database Foundations, Database Programming with SQL

Sr. No.		
2.	Databases: Advanced Topics in SQL https://online.stanford.edu/courses/soe-ydatabases0001-databases-advanced-topics-sql	
3.	Databases: Relational Databases and SQL https://online.stanford.edu/courses/soe-ydatabases0005-databases-relational-databases-and-sql	
4.	Databases: Semistructured Data https://online.stanford.edu/courses/soe-ydatabases0004-databases-semistructured-data	
5.	Data Base Management System, IIT Kharagpur (Prof. Partha Pratim Das Prof. Samiran Chattopadhyay Prof. Kausik Datta) URL: https://nptel.ac.in/courses/106105175	

LIST OF EXPERIMENTS:

Sr. No.	List of Experiments	COs	Total Hours
1	Understand the DBMS Lab environment and Installation. Install and configure MySQL / PostgreSQL / Oracle XE. Set up CLI tools (mysqlsh / psql / SQL*Plus) and GUI clients (MySQL Workbench / pgAdmin / Oracle SQL Developer and Oracle Apex Dashboard).	CO1	2
2	Create a normalized ER Diagram of a real-life database application.	CO 2	2
3	Implement Data Definition Commands: CREATE / ALTER / DROP / RENAME schemas, tables, and rich types (ENUM, JSON, UUID, GENERATED columns)		2
4	Implement SQL code to add / modify constraints — NOT NULL, PRIMARY / FOREIGN KEY with ON UPDATE / DELETE CASCADE, UNIQUE, CHECK, DEFAULT. Validate constraints with sample inserts / updates / deletes.		2
5	Implement SQL code to use Data Manipulation Commands for inserting, deleting, updating and retrieving Tables.		2
6	Implement SELECT queries with WHERE, ORDER BY, DISTINCT, etc clauses. Perform Database Querying using Simple queries and Nested queries, subqueries, EXISTS, IN, and Common Table Expressions (CTEs).	CO3	2
7	Perform Sub-queries and aggregate queries using COUNT, SUM, AVG, MAX, MIN with GROUP BY and HAVING.		2
8	Implement SQL code for performing various types of Joins by connecting different tables: INNER, LEFT, RIGHT, FULL OUTER, CROSS, SELF, and NATURAL JOINs. Join two, three and / or more tables with meaningful relationships.		2

9	Implement SQL code to create, update and drop views. Create updatable and read only views. Build materialized views with manual or automatic refresh.		2
10	Understand use of indices and write SQL code to create index and unique index.		2
11	Implement PL/SQL code to make use of stored procedures and functions with input/output parameters. Include conditional logic (IF, CASE), loops (WHILE, FOR), and exception handling. Use cursors for row-by-row processing in PL/SQL.		2
12	Implement PL/SQL code to create Triggers: (a) before insert (b)after insert (c) before update (d) after update (e) before delete (f) after delete.		2
13	Implement SQL code to carry out Data Control Language (DCL) commands. Practice permission management using GRANT and REVOKE to control access to database objects for different users.	CO4	1
14	Implement SQL code to carry out Transaction control commands. Practice transactional control with COMMIT, ROLLBACK, and SAVEPOINT.		1
15	Carry out project-based learning by creating a project / game / web / desktop application with front end and database connectivity.	CO5	4
	Total hours		**30**

COURSE OUTCOMES WITH PROGRAM OUTCOMES:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	2	-	-	-	-	-	-	1
CO2	3	3	2	-	2	-	-	-	-	1	-	1
CO3	3	3	3	-	3	-	-	-	-	2	-	1
CO4	3	3	2	2	3	-	-	-	-	1	-	1
CO5	3	2	3	2	3	1	1	1	2	2	2	3