GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Engineering

Level: Diploma

Branch: Computer Engineering / Computer Science & Engineering

Subject Code : DI03000031

Subject Name : Relational Database Management System

W.E. F. Academic Year:	2024-2025
Semester:	3rd
Category of the Course:	PCC

Prerequisite:	Basic computer literacy, Logical Thinking and Basic Mathematics
Rationale:	The aim of this course is to get broad understanding of the basic concepts of database management system used for business, scientific and engineering application which stored centralized. The students will develop the skills to develop manage & retrieve data from different perspective using Structured Query Language (SQL) in ORACLE (centralized storage) so there is no need of storing data in files and paper. This will turn reduce of paper wastage. By the end of this course the students will be able to write simple and advanced PL/SQL code blocks, use advanced features such as cursors and bulk fetches and database designing with normalization. Hence students will be able to design database which will be helpful to them in the designing phase of project in the upcoming semester.

Course Outcome:

After Completion of the Course, Student will able to:

No	Course Outcomes	RBT Level
01	Apply database concepts and SQL commands to design, manipulate, and manage relational databases.	Apply
02	Use joins and subqueries to retrieve and manipulate data from multiple tables.	Apply
03	Apply normalization techniques to design efficient database schemas.	Apply
04	Implement integrity constraints to maintain data consistency and accuracy.	Apply
05	Develop PL/SQL programs incorporating control structures, exceptions, cursors, procedures, functions, and triggers for database automation.	Apply

**Revised Bloom's Taxonomy (RBT)*

Teaching and Examination Scheme:

Teaching Scheme (in Hours)			Total Credits L+T+ (PR/2)	Assessment Pattern and Marks				Total Marks
				Theory		Tutorial / Practical		
L	T	PR	C	ESE (E)	PA(M)	PA(I)	ESE (V)	
3	0	2	4	70	30	20	30	150

Course Content:

Unit No.	Content	No. of Hours	% of Weightage
1.	**Introduction to Database System and SQL commands** **Concepts and Definitions** • Data and Information: Definition, differences between data and information • Data Elements: Data items (fields), records, files, and metadata • Data Dictionary: Definition, components, and significance • Database Concepts: Definition, characteristics, and importance of databases • Database Systems and Environment: Components, architecture, and functions • Schemas and Instances: Concepts of schemas, sub-schemas, and instances in a database system **Data Types** • Overview of SQL data types: Numeric, Character, Date/Time **SQL Commands:** • DDL (Data Definition Language): Create Database, Create Table, Alter Table, Drop Table, Truncate Table • DML (Data Manipulation Language): Insert, Select,	12	27

	Update, Delete • DQL (Data Query Language): SELECT statement with WHERE clause, SELECT DISTINCT statement • DCL (Data Control Language): Creating Users & Roles, GRANT, REVOKE commands • TCL (Transaction Control Language): Introduction to Transactions, ACID properties, Commit, Rollback, Savepoint Problem Solving practice on the above concepts		
2.	**SQL In-Built Functions and Joins** **Operators:** • Arithmetic, Comparison, Logical • Special Operators: IN, ANY, ALL, BETWEEN, EXISTS, LIKE Use of System table/DUAL table **SQL Functions:** Single Row Functions: • Numeric Functions: ABS, POWER, MOD, ROUND, TRUNC, SQRT • Character Functions: INITCAP, LOWER, UPPER, LTRIM, RTRIM, REPLACE, SUBSTR, INSTR • Date Functions: ADD_MONTHS, MONTHS_BETWEEN, ROUND, TRUNC • Conversion Functions: TO_CHAR, TO_DATE, TO_NUMBER **Aggregate Functions**: SUM, AVG, COUNT, MAX, MIN **SQL Clauses:** GROUP BY, HAVING, and ORDER BY **Joins:** INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN, CROSS JOIN, SELF JOIN **Subquery:** Nested Queries, Correlated Subqueries	11	24

	Set Operators: MINUS, INTERSECT, UNION, UNION ALL Problem Solving practice on the above concepts		
3.	**Normalization** **Basics of Normalization** • Definition and Importance **Normal Forms** • First Normal Form (1NF) • Second Normal Form (2NF) • Third Normal Form (3NF) **Advantages and disadvantages of Normalization** Problem Solving practice on the above concepts	06	13
4.	**Database Integrity Constraints & Objects** **Integrity Constraints** • Domain Integrity Constraints: NOT NULL, CHECK • Entity Integrity Constraints: UNIQUE, PRIMARY KEY • Referential Integrity Constraints: FOREIGN KEY, REFERENCES, ON DELETE CASCADE **Database Objects** • Views: CREATE VIEW, ALTER VIEW, DROP VIEW • Synonyms: CREATE SYNONYM, DROP SYNONYM • Sequences: CREATE SEQUENCE, ALTER SEQUENCE, DROP SEQUENCE • Indexes: CREATE INDEX (UNIQUE, COMPOSITE), DROP INDEX Problem Solving practice on the above concepts	07	16
5.	**PL/SQL and Triggers** **Fundamentals of PL/SQL** • Basics of PL/SQL • Data Types: %TYPE and %ROWTYPE • Advantages of PL/SQL over SQL **Control Structures**	09	20

	• Conditional, Iterative, and Sequential Statements **Cursors** • Static Cursors: Implicit & Explicit **Procedures and Functions** • Creating and Executing Stored Procedures and Functions **Triggers** • Fundamentals of Triggers • Types of Triggers: BEFORE, AFTER, FOR EACH ROW, FOR EACH STATEMENT Problem Solving practice on the above concepts		
	Total	45	100

Suggested Specification Table with Marks (Theory):

Distribution of Theory Marks (in %)					
R Level	**U Level**	**A Level**	**N Level**	**E Level**	**C Level**
26	**44**	**30**	**0**	**0**	**0**

Where R: Remember; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create (as per Revised Bloom's Taxonomy)

References/Suggested Learning Resources:

(a) Books:

1. Singh, S.K., "Database Systems Concepts, design and Applications 2/e", Pearson Education, New Delhi, Latest Edition
2. Bayross, Ivan, "SQL/PL/SQL", BPB, New Delhi, Latest Edition.
3. Date, C. J., "An Introduction to Database Systems", Pearson Education, New Delhi, Latest Edition
4. Korth, Henry, "Database System Concepts", McGraw-Hill, Delhi, Latest Edition
5. ITLESL, "Introduction to Database Systems", Pearson Education, New Delhi, Latest Edition

(b) Open source software and website:

1. DBMS:http://nptel.iitm.ac.in/video.php?subjectId=106106093
2. SQL Plus Tutorial: http://holowczak.com/oracle-sqlplus-tutorial/
3. Database Tutorials:http://www.roseindia.net/programming-tutorial/Database-Tutorials
4. SQL Basic Concepts: http://www.w3schools.com/sql/
5. SQL Tutorial : http://beginner-sql-tutorial.com/sql.htm

Suggested Course Practical List:

1. Implement SQL queries to perform various DDL commands.
    a. Create at least five tables using different data types and apply various DDL commands (CREATE, ALTER, DROP, TRUNCATE).
2. Implement SQL queries to perform various DML commands.
    a. Insert at least 10 rows using different insert methods.
    b. Edit and remove data using UPDATE and DELETE commands.
    c. Retrieve data using SELECT and various SQL operators.
3. Perform queries for TCL and DCL commands.
    a. Use COMMIT, ROLLBACK, SAVEPOINT for transaction control.
    b. Create users and roles, and apply GRANT, REVOKE commands.
4. Implement SQL queries using date functions.
    a. Functions: ADD_MONTHS, MONTHS_BETWEEN, ROUND, NEXT_DAY, TRUNC, SYSDATE, LAST_DAY, TO_CHAR
5. Implement SQL queries using numeric and character functions.
    a. Numeric functions: ABS, CEIL, POWER, MOD, ROUND, TRUNC, SQRT
    b. Character functions: INITCAP, LOWER, UPPER, LTRIM, RTRIM, REPLACE, SUBSTRING, INSTR
6. Implement SQL queries using conversion and aggregate functions.
    a. Conversion functions: TO_CHAR, TO_DATE, TO_NUMBER
    b. Aggregate functions: AVG, MIN, MAX, SUM, COUNT
    c. Use DECODE function for conditional logic.
7. Implement SQL queries using GROUP BY, HAVING, and ORDER BY clauses.
8. Implement SQL queries using set operators.
    a. UNION, UNION ALL, INTERSECT, MINUS
9. Retrieve data using various types of joins.
    a. INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN, CROSS JOIN, SELF JOIN
10. Retrieve data using subqueries.
    a. Implement multi-level nested subqueries (minimum 3 levels).
    b. Use correlated subqueries for advanced filtering.
11. Re-implement Practical-1 with integrity constraints.
    a. Apply Domain Integrity (NOT NULL, CHECK)
    b. Apply Entity Integrity (UNIQUE, PRIMARY KEY)
    c. Apply Referential Integrity (FOREIGN KEY, ON DELETE CASCADE)
12. Perform queries to create, alter, and drop views.
13. Perform queries to create synonyms, sequences, and indexes.

14. Implement PL/SQL programs using control structures.
    a. Use IF-ELSE, CASE, LOOP, WHILE, FOR statements.
15. Implement PL/SQL programs using cursors.
    a. Work with implicit and explicit cursors.
    b. Use FOR loops with cursors.
16. Implement PL/SQL programs using exception handling.
    a. Handle predefined and user-defined exceptions.
17. Implement user-defined procedures and functions using PL/SQL.
    a. Create and execute stored procedures and functions.
18. Implement various triggers.
    a. Implement BEFORE and AFTER triggers on INSERT, UPDATE, DELETE events.
19. Practice normalization using a sample database.
    a. Apply 1NF, 2NF, 3NF to a database schema.

List of Laboratory/Learning Resources Required:

1. Computer with basic configuration with windows or unix os
2. Text Editor (VS Code, Sublime Text, Atom, Vim or any other editor)
3. Software: SQL/PLSQL supporting software. (e.g.Oracle, SQLServer, MySQL)

Suggested Activities for Students:

1. Case Study on Database Design:
    a. Analyze an existing database system (e.g., banking, e-commerce, hospital management) and suggest improvements using normalization techniques.
2. Mini Project on Database Development:
    a. Design and implement a small database project, such as a student attendance system, library management system, or inventory management system using SQL and PL/SQL.
3. Performance Optimization in SQL Queries:
    a. Compare execution time for queries with and without indexes, joins, and subqueries.
    b. Experiment with query optimization techniques.
4. Data Migration & Backup:
    a. Practice exporting and importing database data using SQL commands (e.g., mysqldump, Oracle Data Pump).
5. Developing a Web-Based Database Application:
    a. Connect a frontend application (using PHP, Python Flask, Node.js) to an SQL database.
6. PL/SQL Performance Tuning:
    a. Analyze performance differences between explicit and implicit cursors.
7. Automating Tasks using Triggers:

a. Implement a trigger-based automation (e.g., audit logging system to track updates and deletions).

8. Gamified Learning:
   a. Conduct SQL query competitions where students solve real-world queries and optimize them for performance.

****