

## GUJARAT TECHNOLOGICAL UNIVERSITY (GTU)

### Competency-focused Outcome-based Green Curriculum-2021 (COGC-2021) Semester-III

#### Course Title: Relational Database Management Systems (Course Code: 4330702)

Diploma programme in which this course is offered	Semester in which offered
Computer Engineering	Third

### 1. 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 centralised. 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.

### 2. COMPETENCY

The course should be taught and implemented with the aim to develop various types of skills so that students are able to acquire following competency:

- **Design, Develop and manage databases for simple applications using Structured Query Language (SQL) and PL/SQL in ORACLE.**

### 3. COURSE OUTCOMES (COs)

The practical exercises, the underpinning knowledge and the relevant soft skills associated with this competency are to be developed in the student to display the following COs:

- a) Perform queries on datasets using SQL\*Plus
- b) Perform joins, subqueries and nested queries on multiple tables using SQL\*plus
- c) Apply rules on datasets using SQL\*Plus constraints
- d) Apply various Normalization techniques.
- e) Write PL/SQL block using concept of Cursor Management, Error Handling, Package and Triggers

### 4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T/2+P/2)	Examination Scheme			
				Theory Marks		Practical Marks	
L	T	P	C	CA	ESE	CA	ESE

3	-	4	5	30*	70	25	25	150
---	---	---	---	-----	----	----	----	-----

(\*): Out of 30 marks under the theory CA, 10 marks are for assessment of the micro-project to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessing the attainment of the cognitive domain UOs required for the attainment of the COs.

**Legends:** **L**-Lecture; **T** – Tutorial/Teacher Guided Theory Practice; **P** -Practical; **C** – Credit, **CA** - Continuous Assessment; **ESE** -End Semester Examination.

## 5. SUGGESTED PRACTICAL EXERCISES

The following practical outcomes (PrOs) that are the sub-components of the COs. *Some of the PrOs marked ‘\*’ are compulsory, as they are crucial for that particular CO at the ‘Precision Level’ of Dave’s Taxonomy related to ‘Psychomotor Domain’.*

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
1	Implement SQL queries to perform various DDL Commands. (Create minimum 5 tables with different data types and operate upon them)	I	02
2	a. Implement SQL queries to perform various DML Commands. (Insert minimum 10 rows using different insert methods, edit and remove data using update and delete commands) b. Retrieve data using SELECT command and various SQL operators.	I	04
3	Perform queries for TCL and DCL Commands	I	02
4	Implement SQL queries using Date functions like add-months, months-between, round, nextday, truncate etc	II	02
5	Implement SQL queries using Numeric functions like abs, ceil, power, mod, round, trunc, sqrt etc. and Character Functions like initcap, lower, upper, ltrim, rtrim, replace, substring, instr etc.	II	02
6	Implement SQL queries using Conversion Functions like to-char, to-date, to-number and Group functions like Avg, Min, Max, Sum, Count, Decode etc.	II	02
7	Implement SQL queries using Group by, Having and Order by clause	II	04
8	Implement SQL queries using simple Case Operations and using Group functions and Case operations for getting summary data	II	02
9	Implement SQL queries using Set operators like Union, unionall, Intersect, Minus etc.	II	02
10	Retrieve data spread across various tables or same table using various Joins.	II	02
11	Retrieve data from multiple tables using Subqueries (Multiple, Correlated) (write minimum 3 level subquery)	II	04
12	Perform queries to Create, alter and update views	III	02
13	Implement Practical-1 again with Domain Integrity, Entity Integrity and Referential Integrity constraints.	III	02
14	Perform queries to Create synonyms, sequence and index	III	02
15	Implement PL/SQL programs using control structures	IV	02
16	Implement PL/SQL programs using Cursors	IV	04
17	Implement PL/SQL programs using exception handling.	IV	02

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. required
18	Implement user defined procedures and functions using PL/SQL blocks	IV	04
19	Perform various operations on packages.	IV	02
20	Implement various triggers	IV	04
21	Draw E-R Diagram of the given problem statements.	V	06
22	Practices on Normalization – using any database perform various normal forms.	V	04
<b>Total</b>			<b>62</b>

**Note**

- i. More **Practical Exercises** can be designed and offered by the respective course teacher to develop the industry relevant skills/outcomes to match the COs. The above table is only a suggestive list.
- ii. The following are some **sample** 'Process' and 'Product' related skills (more may be added/deleted depending on the course) that occur in the above listed **Practical Exercises** of this course required which are embedded in the COs and ultimately the competency.

S. No.	Sample Performance Indicators for the PrOs	Weightage in %
1	Problem Analysis	20
2	Development of the Solution	20
3	Testing of the Solution	10
4	Record observations correctly	20
5	Interpret the result and conclude	30
<b>Total</b>		<b>100</b>

**6. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED**

- a. Hardware: Computer Systems with minimum PIV processor (or equivalent) and 1 GB RAM.
- b. Software: SQL/PLSQL supporting software. (e.g. Oracle, SQLServer, MySQL)

**7. AFFECTIVE DOMAIN OUTCOMES**

The following **sample** Affective Domain Outcomes (ADOs) are embedded in many of the above-mentioned COs and PrOs. More could be added to fulfill the development of this competency.

- a) Work as a leader/a team member.
- b) Follow ethical practices.
- c) Practice environment friendly methods and processes. (Environment related)

The ADOs are best developed through the laboratory/field-based exercises. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

- i. 'Valuing Level' in 1<sup>st</sup> year
- ii. 'Organization Level' in 2<sup>nd</sup> year.
- iii. 'Characterization Level' in 3<sup>rd</sup> year.

## 9. UNDERPINNING THEORY

Only the major Underpinning Theory is formulated as higher level UOs of *Revised Bloom's taxonomy* in order development of the COs and competency is not missed out by the students and teachers. If required, more such higher level UOs could be included by the course teacher to focus on attainment of COs and competency.

Unit	Unit Outcomes (UOs) (4 to 6 UOs at Application and above level)	Topics and Sub-topics
<b>Unit – I Introduction to Database System and SQL commands</b>	1a. Differentiate the terms: Data, Information, Records, Fields, Metadata, Data warehouse, Data dictionary	1.1 Concepts and Definitions: Database and database systems and database environment 1.2 Data, Information, Data Item or Fields, Records, Files, Metadata, Data dictionary and it's components, 1.3 Schemas, Sub-schemas, and Instances
	1b. DBMS Data types, Creating Tables (DDL), Managing Tables (DML) with SQL,	1.4 Data types 1.5 Database Language commands: Data Definition Language (DDL): CREATE, ALTER, TRUNCATE, DROP 1.6 Database Language: Data Manipulation Language (DML): INSERT, SELECT, UPDATE, DELETE
	1c. Describe & practice Transaction Control Data Control Language	1.7 Transactional Control: Commit, Save point, Rollback 1.8 DCL Commands: Grant and Revoke
<b>Unit – II SQL In built functions and Joins</b>	2a. Execute various SQL operators and Functions	2.1 Operators Arithmetic, Comparison, Logical SQL functions- Single row function i. Single row function. ii. Date functions (add-months, months-between, round, truncate). iii. Numeric Functions (abs, power, mod, round, trunc, sqrt) iv. Character Functions (initcap, lower, upper, ltrim, rtrim, replace, substring, instr) v. Conversion Functions (to-char, to-date, to-number)
	2b. Perform queries on 'Group by', 'Having' and 'Order by' clause	2.2 Groupby, Having and Order by clause
	2c. Implement 'Joins'	2.3 Joins: Simple, Equi-join, Non-equi, Self-Joins, Outer-joins.

<b>Unit</b>	<b>Unit Outcomes (UOs)</b> (4 to 6 UOs at Application and above level)	<b>Topics and Sub-topics</b>
	2d. Perform different types of sub queries	2.4 Subqueries - Multiple, Correlated 2.5 Implementation of Queries using SQL Set operators: Union, union all, Intersect, Minus
<b>Unit– III Database Integrity Constraints &amp; Objects</b>	3a. Describe with examples Domain Integrity and Entity Integrity constraint	3.1 Domain Integrity constraints: Not null, Check 3.2 Entity Integrity constraints: Unique, Primary key.
	3b. Describe with examples Referential Integrity constraint	3.3 Referential Integrity constraints: Foreign key, referenced key, on delete cascade
	3c. Database Objects	3.4 Views – Create, Alter, Drop views 3.5 Synonym: Create, Drop synonym 3.6 Sequences: Create, alter, Drop sequences 3.7 Index: Unique and composite – Create, Drop
<b>Unit– IV PL/ SQL and Triggers</b>	4a. Describe the fundamentals of the PL/SQL programming language	4.1 Basics of PL / SQL 4.1.1 Data types 4.2 Advantages of PL/SQL over SQL
	4b. Use different Control Structures 4c. Write and execute PL/SQL programs in SQL*Plus	4.3 Control Structures: Conditional, Iterative, Sequential
	4d. Describe & Implement Concepts of exception handling	4.4 Exceptions: Predefined Exceptions, User defined exceptions
	4e. Implement cursor, procedure and function in Package	4.5 Cursors: Static (Implicit & Explicit), Dynamic 4.6 Procedures & Functions
	4f. Describe the various types of triggers 4g. Write, code, test and debug various types of triggers	4.7 Fundamentals of Database Triggers 4.8 Creating Triggers 4.9 Types of Triggers: Before, after for each row, for each statement
<b>Unit– V Normalization</b>	5a. Describe different Normal Forms 5b. Solve problems of normalization 5c. Describe advantages and disadvantages of Normalization	5.1 Basics of Normalization 5.2 Normal Forms 4.2.1 First Normal Form(1NF) 4.2.2 Second Normal Form(2NF) 4.2.3 Third Normal Form(3NF) 5.3 Advantages and disadvantages of Normalization

*Note: The UOs need to be formulated at the 'Application Level' and above of Revised Bloom's Taxonomy' to accelerate the attainment of the COs and the competency.*

**10. SUGGESTED SPECIFICATION TABLE FOR QUESTIONPAPER DESIGN**

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Introduction to Database System and SQL commands	10	7	5	4	16
II	SQL In built functions and Joins	8	5	4	5	14
III	Database Integrity Constraints & Objects	8	5	4	5	14
IV	PL/ SQL and Triggers	10	6	5	5	16
V	Normalization	6	4	3	3	10
<b>Total</b>		<b>42</b>	<b>27</b>	<b>21</b>	<b>22</b>	<b>70</b>

**Legends:** R=Remember, U=Understand, A=Apply and above (Revised Bloom's taxonomy)

**Note:** This specification table provides general guidelines to assist student for their learning and to teachers to teach and question paper designers/setters to formulate test items/questions assess the attainment of the UOs. The actual distribution of marks at different taxonomy levels (of R, U and A) in the question paper may vary slightly from above table.

**11. SUGGESTED STUDENT ACTIVITIES**

Other than the classroom and laboratory learning, following are the suggested student-related **co-curricular** activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:

- i. Prepare seminar presentations explaining the organization of database in various live systems like banking, insurance, online booking etc.
- ii. Prepare power point presentation for different database objects.
- iii. Prepare case study explaining the need for converting a large table to many smaller tables using 1NF, 2NF, 3NF.
- iv. Design database which can be used in the course on .Net programming

**12. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)**

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

- a) Massive open online courses (**MOOCs**) may be used to teach various topics/sub topics.
- b) Guide student(s) in undertaking micro-projects.
- c) '**L**' in section No. 4 means different types of teaching methods that are to be employed by teachers to develop the outcomes.
- d) About **20% of the topics/sub-topics** which are relatively simpler or descriptive in nature is to be given to the students for **self-learning**, but to be assessed using different assessment methods.
- e) With respect to **section No.11**, teachers need to ensure to create opportunities and provisions for **co-curricular activities**.
- f) Guide students on how to address issues on environ and sustainability

### 13. SUGGESTED MICRO-PROJECTS

**Only one micro-project** is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project is group-based. However, in the fifth and sixth semesters, it should be preferably be individually undertaken to build up the skill and confidence in every student to become problem solver so that s/he contributes to the projects of the industry. In special situations where groups have to be formed for micro-projects, the number of students in the group should **not exceed three**.

The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total duration of the micro-project should not be less than **16 (sixteen) student engagement hours** during the course. The student ought to submit micro-project by the end of the semester to develop the industry-oriented COs.

A suggestive list of micro-projects is given here. This has to match the competency and the COs. Similar micro-projects could be added by the concerned course teacher:

- Do related activities such as prepare E-R Model, Relational Model, do Normalization, Create Tables, insert data, Delete Data, Query database, create stored procedures, etc.
  - a) **Effective Library Management System:** Book, student and faculty record with issuance, retrieval and fine details
  - b) **Payroll:** Useful for all organizations as every organization definitely has employees. No of days worked, salary, leaves etc. can be kept.
  - c) **University:** Details of department and its courses. Track of students, teachers and staff can also be kept.
  - d) **IT TRAINING GROUP DATABASE PROJECT:** This can be clearly indicated with an ER diagram. Entities, Relationships and Key constraints should be clearly indicated. The company has 50 instructors and can handle 500 trainees for each training session. There are 20 advanced technology courses in all that company offers. Each course is well handled by a team of 5 or more trainers. Maximum of 3 teaching teams is assigned to every instructor. They might also be asked to do some research. One advanced technology course is handled by per trainee per session.
  - e) **Employee database:** ‘Create’ employee table, ‘Select’ and display an employee matching a given condition, ‘Delete’ duplicate records, delete rows using triggers, insert and update records, find net salary, etc.
  - f) Case Study: **Visitor Management database**
  - g) Case Study: **Students Academic database**
  - h) Case Study: **Inventory Management System database**
  - i) Case study: **Bank Operations database**
  - j) Case Study: **Bus Operator (Roadways)**

### 14. SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication with place, year and ISBN
1	SQL/PL/SQL	Bayross, Ivan	BPB, NewDelhi, 2010.
2	Database Systems Concepts, design and	Singh, S.K.	Pearson Education, New Delhi,2011

S. No.	Title of Book	Author	Publication with place, year and ISBN
	Applications 2/e		
3	An Introduction to Database Systems	Date, C. J.	Pearson Education, New Delhi,2006
4	Database System Concepts,	Korth, Henry	McGrawHill,Delhi,2011
5	Introduction to Database Systems	ITLES.L.	Pearson Education, New Delhi,2010

### 15. SOFTWARE/LEARNING WEBSITES

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

### 16. PO-COMPETENCY-CO MAPPING

Semester III	Relational Database Management Systems (Course Code: .....									
	POs and PSOs									
Competency & Course Outcomes	PO 1 Basic & Discipline specific knowledge	PO 2 Problem Analysis	PO 3 Design/development of solutions	PO 4 Engineering Tools, Experimentation & Testing	PO 5 Engineering practices for society, sustainability & environment	PO 6 Project Management	PO 7 Life-long learning	PSO 1	PSO 2	PSO 3 (If needed)
<b>Competency</b> Design, Develop and manage databases for simple applications using Structured Query Language (SQL) and PL/SQL in ORACLE										
<b>Course Outcomes</b> CO a) Perform queries on datasets using SQL*Plus	3	-	-	-	1	-	1	-	-	
CO b) Perform joins, sub queries and nested queries on multiple tables using SQL*plus	3	2	1	2	1	-	1	-	-	
CO c) Apply rules on datasets using SQL*Plus constraints	2	3	1	-	2	2	1	1	-	
CO d) Apply various Normalization techniques.	3	1	1	1	1	2	1	1	-	
CO e) Perform PL/SQL programming using concept of Cursor Management, Error Handling, Package and Triggers.	3	2	2	1	1	1	1	-	-	



Legend: '3' for high, '2' for medium, '1' for low or '-' for the relevant correlation of each competency, CO, with PO/ PSO

## 17. COURSE CURRICULUM DEVELOPMENT COMMITTEE

### GTU Resource Persons

S. No.	Name and Designation	Institute	Contact No.	Email
1	<b>Prof. B.H. Kantewala</b> , Head Department of Computer Engineering	CU Shah Polytechnic, Surendranagar	9428000592	bhkantevala@yahoo. com
2	<b>Prof. M. P. Mehta</b> , Head Department of Computer Engineering	Govt. Polytechnic, Gandhinagar	9879578273	manishamehtain@g mail.com
3	<b>Prof. Sachin D. Shah</b> , Lecturer Department of Computer Engineering	RCTI, Ahmedabad	9427955671	sachindshah@yahoo. com
4	<b>Miss Dhara H. Wagh</b> Lecturer Department of Computer Engineering	Govt. Polytechnic, Gandhinagar	9427465614	dhara.wagh1791@g mail.com
5	<b>Ms. Pravina R. Mehta</b> , Lecturer Department of Computer Engineering.	Govt. Polytechnic, Himatnagar	9825446175	pravina6mehta@gm ail.com