CE246: Database Management System

Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	4	4	-	8	6
Marks	100	100	-	200	

Pre-requisite courses:

- Data Structure

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Introductory concepts of DBMS	04
2.	Relational Model	06
3.	Entity-Relationship model	07
4.	Formal Relational Query Languages	06
5.	Relational Database Design	09
6.	Transaction & Recovery Management	07
7.	Advanced Transaction Processing	06
8.	Database Security	06
9.	Indexing and Hashing	07
10.	Query Processing & Query Optimization	02
	Total hours (Theory) :	60
	Total hours (Lab) :	60
	Total hours :	120

Detailed Syllabus:

1.	**Introductory concepts of DBMS**	**04 Hours**	**06%**
	Introduction and applications of DBMS, Purpose of database, Data Independence, Database System architecture- levels, Mappings, Database users and DBA		
2.	**Relational Model**	**06 Hours**	**10%**
	Structure of Relational Databases, Database Schema, Schema Diagram, Domains , Relations, Relational Query Languages, Relational Operations		

3.	**Entity-Relationship model**	**07 Hours**	**12%**
	Basic concepts, Design process, Constraints, Keys, Design issues, E-R diagrams, Weak Entity Sets, Extended E-R features-Generalization, Specialization, Aggregation, Reduction to E-R database schema		
4.	**Formal Relational Query Languages**	**06 Hours**	**10%**
	The relational Algebra, The Tuple Relational Calculus, The Domain Relational Calculus		
5.	**Relational Database design**	**09 Hours**	**15%**
	Functional Dependency–definition, Trivial and Non-Trivial FD, Closure of FD set, Closure of attributes, Irreducible set of FD, Normalization – 1NF, 2NF,3NF, Decomposition using FD-Dependency Preservation		
6.	**Transaction & Recovery Management**	**07 Hours**	**12%**
	Transaction concepts, Properties of Transactions, Serializability of transactions, Testing for Serializability, System recovery, Two- Phase Commit protocol, Recovery and Atomicity, Log-based recovery, Concurrent executions of transactions and related problems, Locking mechanism, Solution to Concurrency Related Problems, Deadlock, Two-phase locking protocol, Intent locking		
7.	**Advanced Transaction Processing**	**06 Hours**	**10%**
	Transaction-Processing Monitors, Transactional Workflows, Main-Memory Databases, Real-Time Transaction Systems, Long-Duration Transactions		
8.	**Database Security**	**06 Hours**	**10%**
	Views - What are views for?, View retrievals, View updates, Snapshots (a digression), Materialized view, Security – Security and Authentication, authorization in SQL, Data encryption, Missing Information - An overview of the 3VL approach		
9.	**Indexing and Hashing**	**07 Hours**	**12%**
	Basic Concepts, Ordered Indices, B+-Tree Index Files, B+-Tree Extensions, Multiple-Key Access, Static Hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices, Index Definition in SQL		
10.	**Query Processing & Query Optimization**	**02 Hours**	**03%**

	Overview, Measures of Query Cost, Selection Operation, Sorting, Join, Evaluation of Expressions, Transformation of relational Expressions, Estimating Statistics of expression results, Query Evaluation plans		

Course Outcomes (COs):

At the end of the course, students will be able to

CO1	Apply the concepts of engineering i.e collecting data, organize the data in the systematic form, arrange the data in a computational way and applying mathematics formation.
CO2	Analyse how data are stored and maintained using data models. Ready to assimilate the concept of data abstraction and design queries using SQL. Identify how data is represented in the relational model and create relations using SQL language
CO3	Identify and evaluate the constructs in the E-R model and issues involved in developing an E-R diagram. Convert an E-R diagram into a relational database schema. Declare and enforce integrity constraints on database using a state-of-art RDBMS.
CO4	Produce aggregate operators to write SQL queries which are not expressible in relational algebra. “More mathematical” notation may apply and also used in research and other venues. Combining these concepts allows production of sophisticated queries.
CO5	Decompose un-normalized tables into normalized compliant tables. Design and implement a normalize database schema for a given problem-domain.
CO6	Compare transactions and their properties with (ACID) and without ACID. Apply locking protocol to ensure isolation. Develop logging technique to ensure atomicity and durability
CO7	Design a logical view which can be used for analytical tasks. Develop practical experience of the design and implement scalable, secure databases.
CO8	Produce strategies to minimise risks of security breaches in a range of network environments and data storage systems. Compute retrieval time and concluding with suitable indexing technique
CO9	Compare and evaluate query execution plan.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	2	1	2	1	1	-	-	-	1	2	-
CO2	3	3	3	2	3	1	2	-	-	-	-	3	2	1
CO3	3	3	3	3	2	2	1	3	2	1	-	2	3	2
CO4	3	3	1	3	1	1	-	2	2	-	-	2	2	2
CO5	2	2	3	2	3	1	2	2	1	-	-	3	1	1
CO6	3	3	2	1	-	2	-	-	1	-	-	2	-	1
CO7	3	3	3	2	3	1	-	2	2	1	-	3	2	2
CO8	2	3	2	2	2	2	2	3	1	-	-	3	3	1
CO9	2	2	3	1	3	-	-	1	1	-	-	3	3	1

Enter correlation levels 1, 2 or 3 as defined below:

1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

If there is no correlation, put "**-**"

Recommended Study Material:

- **Text book:**
  1. Database System Concepts, Abraham Silberschatz, Henry F. Korth& S. Sudarshan, McGraw Hill.
  2. An introduction to Database Systems, C J Date, Addition-Wesley
- **Reference book:**
  1. "Fundamentals of Database Systems", R. Elmasri and S.B. Navathe, the Benjamin / Cumming Pub. Co
  2. SQL,PL/SQL the Programming Language of oracle, Ivan Bayross, BPB Publications
  3. Oracle: The Complete Reference, George Koch, Kevin Loney, TMH /oracle press
- **Web material:**
  1. http://www.sql.org
  2. http://www.w3schools.com
  3. http://www.sqlcourse.com
- **Software:**
  1. Oracle 10g
  2. SQL Lite

3. Live SQL
4. Firebase
5. Squirrel SQL
6. Postgre SQL