

**Relational Data Base Design:** Functional Dependencies & Normalization for Relational Databases, Functional Dependencies, Normal Forms Based on Primary Keys, (1NF, 2NF, 3NF & BCNF), Lossless Join and Dependency Preserving Decomposition, Functional dependencies and its closure, covers and equivalence.

#### **UNIT 4**

**Transaction Management:** Transaction Concept and State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability: Testing of serializability, Serializability of schedules, conflict & view serializable schedule.

**Concurrency Control Techniques:** Lock-Based Protocols, Timestamp-based Protocols, validation based protocol. Deadlock Handling

#### **UNIT 5**

##### **Recovery System**

Recoverability: Failure Classification, Storage Structure, Recovery and Atomicity, Log-based Recovery, Shadow Paging, Recovery with Concurrent Transactions

**Storage organization :** Indexing, Hashing ,file storage.

#### **List of Experiments:**

**Q 1:** Consider the following relational schema

SAILORS (sid, sname, rating, date\_of\_birth)

BOATS (bid, bname, color)

RESERVES (sid, bid, date, time slot)

**Write the following queries in SQL and relational algebra**

- 1) Find sailors who've reserved at least one boat
- 2) Find names of sailors who've reserved a red or a green boat in the month of March.
- 3) Find names of sailors who've reserved a red and a green boat
- 4) Find sid of sailors who have not reserved a boat after Jan 2018.
- 5) Find sailors whose rating is greater than that of all the sailors named "John"
- 6) Find sailors who've reserved all boats
- 7) Find name and age of the oldest sailor(s)
- 8) Find the age of the youngest sailor for each rating with at least 2 such sailors

**Q2.** Consider the following relational schema:

CUSTOMER (cust\_num, cust\_lname, cust\_fname, cust\_balance);

PRODUCT (prod\_num, prod\_name, price)

INVOICE (inv\_num, prod\_num, cust\_num, inv\_date, unit\_sold, inv\_amount);

**Write SQL queries and relational algebraic expression for the following**

- 1) Find the names of the customer who have purchased no item. Set default value of Cust\_balance as 0 for such customers.
- 2) Write the trigger to update the CUST\_BALANCE in the CUSTOMER table when a new invoice record is entered for the customer.
- 3) Find the customers who have purchased more than three units of a product on a day.
- 4) Write a query to illustrate Left Outer, Right Outer and Full Outer Join.
- 5) Count number of products sold on each date.
- 6) As soon as customer balance becomes greater than Rs. 100,000, copy the customer\_num in new table called "GOLD\_CUSTOMER"
- 7) Add a new attribute CUST\_DOB in customer table

**Q 3:** Consider the following relational schema

DEPARTMENT(Department\_ID, Name, Location\_ID)

JOB (Job\_ID , Function )

EMPLOYEE (Employee\_ID, name, DOB, Job\_ID , Manager\_ID, Hire\_Date, Salary, department\_id)

**Answer the following queries using SQL and relational algebra:**

- 1) Write a query to count number of employees who joined in March 2015
- 2) Display the Nth highest salary drawing employee details.
- 3) Find the budget (total salary) of each department.
- 4) Find the department with maximum budget.
- 5) Create a view to show number of employees working in Delhi and update it automatically when the database is modified.
- 6) Write a trigger to ensure that no employee of age less than 25 can be inserted in the database.

**Q4: PROJECT**

Students are required to develop a DBMS for the applications assigned to them. Following items are required to be submitted for the project

- a) Problem Statement
- b) ER model/ Relational Model
- c) Integrity Constraints implemented
- d) Suitable Queries to create and manage database

**Note: Students have to make sure that they have defined proper integrity constraints to ensure consistency of database used in assignments as well as project.**

\*\*\*Project is to be carried out in groups of three students.

**SUGGESTED READINGS:**

Text book:

1. Korth, Silbertz, Sudarshan, "Data base concepts", McGraw-Hill.

Reference books

1. Elmasri, Navathe, "Fundamentals of Database systems", Addison-Wesley.
2. Date C.J., "An Introduction to Database systems".