B.Tech. (III Sem.)	20CS56 - DATABASE MANAGEMENT SYSTEMS LAB	L	T	P	Cr.
		0	0	3	1.5

Pre-requisite : Programming language, Discrete Mathematical Structures, and Data Structures.

Course Educational Objective: The objective of this lab is to provide a strong formal foundation in database concepts, technology, and practice to the participants to groom them into well-informed database application developers.

Course Outcomes (CO): *At the end of this course, the student will be able to:*

CO1: Create & manipulate the relational database using SQL.**(Apply- L3)**

CO2: Implement Views, procedures, triggers, and cursors on relational database. **(Apply- L3)**

CO3: Create Unstructured Databases using MongoDB.**(Apply- L3)**

CO 4: Improve individual / teamwork skills, communication & report writing skills with ethical values.

Introduction: Language basics and example queries (**one or two weeks**)

1) Create a table STUDENT with appropriate data types and perform the following queries. Attributes are Roll number, student name, date of birth, branch and year of study.

a) Insert 5 to 10 rows in a table?

b) List all the students of all branches
c) List student names whose name starts with 's'.
d) List student names whose name contains 's as third literal.
e) List student names whose contains two 's' anywhere in the name
f) List students whose branch is NULL.
g) List students of CSE & ECE who born after 1980.
h) List all students in reverse order of their names.
i) Delete students of any branch whose name starts with 's'.
j) Update the branch of CSE students to ECE.
k) Display student name padded with *'after the name of all the students.

2) Create the following tables based on the above Schema Diagram with appropriate data types
and constraints and perform the following queries.

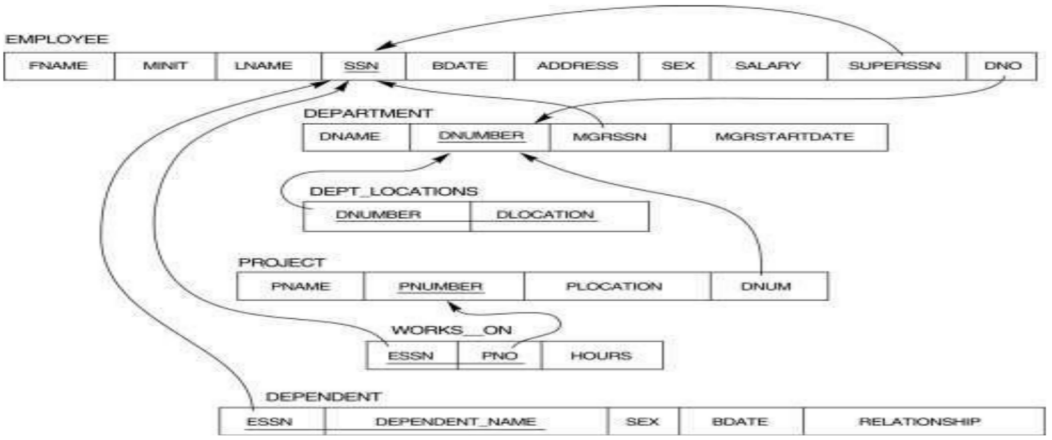
SAILORS (Saild, Salname, Rating, Age)
RESERVES (Sailid, boatid, Day)
BOATS (Boatid, Boat-name, Color)

a) Insert 5 to 10 rows in all tables?

b) Find the name of sailors who reserved boat number 3.
c) Find the name of sailors who reserved green boat.
d) Find the colors of boats reserved by Ramesh.
e) Find the names of sailors who have reserved at least one boat.

f) Find the allsailid of sailors who have a rating of 10 or have reserved boated 104.
g) Find the Sailid's of sailors with age over 20 who have not registered a red boat.
h) Find the names of sailors who have reserved a red or green boat.
i) Find sailors whose rating is better than some sailor called Salvador.
j) Find the names of sailors who are older than the oldest sailor with a rating of 10.

3) Schema Diagram for the rest of the SQL and PLSQL Programs.



Create the following tables based on the above Schema Diagram with appropriate data types and constraints.

EMPLOYEE (Fname, Mname, Lname, SSN, Bdate, Address, Gender, Salary, SuperSSN,Dno)

DEPARTMENT (Dnumber, Dname, MgrSSN, Mgrstartdate)

DEPENDENT (ESSN, Dependent_Name, Gender, Bdate, Relationship)

a) Insert 5 to 10 rows into all the tables.
b) Display all employees' names along with their department names.
c) Display all employees' names along with their dependent details.
d) Display name and address of all employees who work for Research department.
e) List the names of all employees with two or more dependents.
f) List the names of employee who have no dependents.
g) List the names of employees who have at least one dependent.
h) List the names of the employees along with names of their supervisors using aliases.
i) Display name of the department and name of manager for all the departments.
j) Display the name of each employee who has a dependent with the same first name and gender as the employee.

4) Create the following tables based on the above Schema Diagram with appropriate data typesand constraints in addition to the tables in Experiment 2.

DEPT_LOCATIONS (Dnumber, Dloaction)
PROJECT (Pname, Pnumber, Plocation,
Dnum) WORKS_ON (ESSN, Pno, Hours)

a) Insert 5 to 10 rows into all the tables.

b) Find the names of the employees who work on all the projects controlled by the department Research.
c) List the project number, name and no. Of employees who work on that project for all the projects.
d) List the names of all the projects controlled by the departments department wise.
e) Retrieve the names of employees who work on all projects that John works on.
f) List the project numbers for projects that involve an employee either as worker or as a manager of the department that controls the project.
g) List the names of all employees in one department who work more than 10 hours on one specific project.
h) For each project, list the project name and total hours (by all employees) spent on that project.
i) Retrieve the names of all employees who work on every project.
j) Retrieve the names of all employees who do not work on any project.

5) Create a view that has project name, controlling department name, number of employees andtotal hours worked on the project for each project with more than one employee working on it.
   a) List the projects that are controlled by one department from this view.
   b) List the managers of the controlling departments for all the projects.
   c) Demonstrate one update operation on this view.
   d) List the Location of the controlling departments for all the projects.
   e) Retrieve the data from the view.
6) Create a view emp from employee such that it contains only emp_noemp_name and department.
7) Create a view dept from department with only dept_no and location.
8) Create a view that contains the details of employees who are managers only.
9) Write a procedure to check whether the given number is Armstrong or not.
10) Write a procedure which accept the account number of a customer and retrieve the balance.
11) Write a procedure which accepts the student number and displays the department in which he belongs to.
12) Create a cursor to modify the salary of all employees belonging to 'Research' department by 150%.
13) Consider the college database. Retrieve all students who have registered for a specific course and store their details into another table using Cursors.
14) Write an update trigger on Account table. The system should keep track of the records that are being updated.
15) Create NoSQL database for a sample application and perform CURD operations

Design Database for any one of the following Case Studies

Case Study1 : Hospital Management System

Aim: XYZ hospital is a multi-specialty hospital that includes a number of departments,rooms, doctors, nurses, compounders, and other staff working in the hospital. Patients havingdifferent kinds of ailments come to the hospital and get checkup done from the concerneddoctors. If required they are admitted in the hospital and discharged after treatment.The aim of this case study is to design and develop a database for the hospital to maintain therecords of various departments, rooms, and doctors in the hospital. It also maintains recordsof the regular patients, patients admitted in the hospital, the checkup of patients done by thedoctors, the patients that have been operated, and patients discharged from the hospital.

Description: In hospital, there are many departments like Orthopedic, Pathology,Emergency, Dental, Gynecology, Anesthetics, I.C.U., Blood Bank, Operation Theater ,Laboratory, M.R.I., Neurology, Cardiology, Cancer Department, Corpse, etc. There is an OPD where patients come and get a card (that is, entry card of the patient) for check up from the concerned doctor. After making entry in the card, they go to the concerned doctor's room and the doctor checks up their ailments. According to the ailments, the doctor either prescribes medicine or admits the patient in the concerned department. The patient may choose either private or general room according to his/her need. But before getting admission in the hospital, the patient has to fulfill certain formalities of the hospital like room charges,etc. After the treatment is completed, the doctor discharges the patient. Before discharging from the hospital, the patient again has to complete certain formalities of the hospital like balance charges, test charges, operation charges (if any), blood charges, doctors' charges, etc. Next we talk about the doctors of the hospital. There are two types of the doctors in the hospital, namely, regular doctors and call on doctors. Regular doctors are those doctors who come to the hospital daily. Calls on doctors are those doctors who are called by the hospital if the concerned doctor is not available.

Table Description:

Following are the tables along with constraints used in Hospital Management database

1. DEPARTMENT: This table consists of details about the various departments in the hospital. The information stored in this table includes department name, department location, and facilities available in that department.
Constraint: Department name will be unique for each department.

2. ALL_DOCTORS: This table stores information about all the doctors working for the hospital and the departments they are associated with. Each doctor is given an identity number starting with DR or DC prefixes only.

Constraint: Identity number is unique for each doctor and the corresponding department should exist in DEPARTMENT table.

3. DOC_REG: This table stores details of regular doctors working in the hospital. Doctors are referred to by their doctor number. This table also stores personal details of doctors like name, qualification, address, phone number, salary, date of joining, etc. Constraint: Doctor's number entered should contain DR only as a prefix and must exist in ALL_DOCTORS table.

4. DOC_ON_CALL: This table stores details of doctors called by hospital when additional doctors are required. Doctors are referred to by their doctor number. Other personal details like name, qualification, fees per call, payment due, address, phone number, etc., are also stored.

Constraint: Doctor's number entered should contain DC only as a prefix and must existin ALL_DOCTORS table.

5. PAT_ENTRY : The record in this table is created when any patient arrives in thehospital for a checkup. When patient arrives, a patient number is generated which actsas a primary key. Other details like name, age, sex, address, city, phone number, entrydate, name of the doctor referred to, diagnosis, and department name are also stored.After storing the necessary details patient is sent to the doctor for checkup.

Constraint: Patient number should begin with prefix PT. Sex should be M or F only.Doctor's name and department referred must exist.

6. PAT_CHKUP: This table stores the details about the patients who get treatmentfrom the doctor referred to. Details like patient number from patient entry table, doctornumber, date of checkup, diagnosis, and treatment are stored. One more field status isused to indicate whether patient is admitted, referred for operation or is a regularpatient to the hospital. If patient is admitted, further details are stored inPAT_ADMITtable. If patient is referred for operation, the further details are stored in PAT_OPR table and if patient is a regular patient to the hospital, the further details are stored inPAT_REG table.
Constraint: Patient number should exist in PAT_ENTRY table and it should be unique.

7. PAT_ADMIT: When patient is admitted, his/her related details are stored in this table. Information stored includes patient number, advance payment, mode of payment, room number, department, date of admission, initial condition, diagnosis, treatment, number of the doctor under whom treatment is done, attendant name, etc.

Constraint: Patient number should exist in PAT_ENTRY table. Department, doctornumber, room number must be valid.

8. PAT_DIS: An entry is made in this table whenever a patient gets discharged from the hospital. Each entry includes details like patient number, treatment given, treatment advice, payment made, mode of payment, date of discharge, etc.
Constraint: Patient number should exist in PAT_ENTRY table.

9. PAT_REG: Details of regular patients are stored in this table. Information stored includes date of visit, diagnosis, treatment, medicine recommended, status of treatment, etc.

Constraint: Patient number should exist in patient entry table. There can be multiple entries of one patient as patient might be visiting hospital repeatedly for checkup and there will be entry for patient's each visit.

10. PAT_OPR: If patient is operated in the hospital, his/her details are stored in this table. Information stored includes patient number, date of admission, date of operation, number of the doctor who conducted the operation, number of the operation theater in which operation was carried out, type of operation, patient's condition before and after operation, treatment advice, etc.

Constraint: Patient number should exist in PAT_ENTRY table. Department, doctor number should exist or should be valid.

11. ROOM_DETAILS: It contains details of all rooms in the hospital. The details stored in this table include room number, room type (general or private), status (whether occupied or not), if occupied, then patient number, patient name, charges per day, etc.

Constraint: Room number should be unique. Room type can only be G or P and status can only be Y or N.

Case Study2 : Railway Reservation

Aim: The railway reservation system facilitates the passengers to enquire about the trains available on the basis of source and destination, booking and cancellation of tickets, enquire about the status of the booked ticket, etc. The aim of case study is to design and develop a database maintaining the records of different trains, train status, and passengers. The record of train includes its number, name, source, destination, and days on which it is available, whereas record of train status includes dates for which tickets can be booked, total number of seats available, and number of seats already booked. The database has been developed and tested on the Oracle.

Description:

Passengers can book their tickets for the train in which seats are available. For this, passenger must provide the desired train number and the date for which ticket is to be booked. Before booking a ticket for a passenger, the validity of train number and booking date is checked. Once the train number and booking date are validated, it is checked whether the seat is available. If yes, the ticket is booked with confirm status and corresponding ticket IDis generated which is stored along with other details of the passenger. After all the available tickets are booked, certain numbers of tickets are booked with waiting status. If waiting lot is also finished, then tickets are not booked and a message of non-availability of seats is displayed.The ticket once booked can be cancelled at any time. For this, the passenger must provide the ticket ID (the unique key). The ticket ID is searched, and the corresponding record is deleted. With this, the first ticket with waiting status also gets confirmed.

List of Assumption

Since the reservation system is very large in reality, it is not feasible to develop the case study to that extent and prepare documentation at that level. Therefore, a small sample case study has been created to demonstrate the working of the reservation system. To implement this sample case study, some assumptions have been made, which are as follows:

1. The number of trains has been restricted to 5.

2. The booking is open only for next seven days from the current date.

3. Only two categories of tickets can be booked, namely, AC and General.

4. The total number of tickets that can be booked in each category (AC and General) is 10.

5. The total number of tickets that can be given the status of waiting is 2.

6. The in-between stoppage stations and their bookings are not considered.

Description of Tables and Procedures

Tables and procedures that will be created are as follows:

1.TrainList: This table consists of details about all the available trains. The information stored in this table includes train number, train name, source, destination, fair for AC ticket, fair for general ticket, and weekdays on which train is available.

Constraint: The train number is unique.

2. Train_Status: This table consists of details about the dates on which ticket can be booked for a train and the status of the availability of tickets. The information stored in this table includes

train number, train date, total number of AC seats, total number of general seats, number of AC seats booked, and number of general seats booked.

Constraint: Train number should exist in TrainList table.

3. Passenger: This table consists of details about the booked tickets. The information stored in this table includes ticket ID, train number, date for which ticket is booked, name, age, sex and address of the passenger, status of reservation (either confirmed or waiting), and category for which ticket is booked.

Constraint: Ticket ID is unique and the train number should exist in TrainList table.

4.Booking: In this procedure, the train number, train date, and category is read from the passenger. On the basis of the values provided by the passenger, corresponding record is retrieved from the Train_Status table. If the desired category is AC, then total number of AC seats and number of booked AC seats are compared in order to find whether ticket can be booked or not. Similarly, it can be checked for the general category. If ticket can be booked, then passenger details are read and stored in the Passenger table.

5.Cancel: In this procedure, ticket ID is read from the passenger and corresponding record is searched in the Passenger table. If the record exists, it is deleted from the table. After deleting the record (if it is confirmed), first record with waiting status for the same train and same category are searched from the Passenger table and its status is changed to confirm.

Case Study3 :Painting Hire Business

System Description:

A local businesswoman has decided to start her own Internet business,called MasterpiecesLtd, hiring paintings to private individuals and commercial companies.Because of your reputation as a database designer, she has called upon your services todesign and implement a database to support her new business. At the initial planning meeting,to discuss the design, the following user requirements were requested.The system must be able to manage the details of customers, paintings and those paintingscurrently on hire to customers. Customers are categorized as B (bronze), S (silver), G (gold) orP (platinum). These categories entitle a customer to a discount of 0%, 5%, 10% or 15%respectively.

Customers often request paintings by a particular artist or theme (e.g. animal, landscape,seascape, naval, still-life, etc). Over time a customer may hire the same painting morethan once.

Each painting is allocated a customer monthly rental price defined by the owner. The owner of the painting is then paid 10% of that customer rental price. Any paintings that are not hired within six months are returned to the owner. However, after three months, an owner may resubmit a returned painting.Each painting can only have one artist associated with it.Several reports are required from the system. Three main ones are:

1. For each customer, a report showing an overview of all the paintings they have hired or are currently hiring
2. For each artist, a report of all paintings submitted for hire
3. For each artist, a returns report for those paintings not hired over the past six months remember to identify key attributes and any foreign key attributes.