Government Engineering College Painavu
Department of Computer Science and Engineering
CSL 333 DBMS Lab

July 20, 2025

1 SQL Experiments

1. Create the following two tables:

   - **College**: consisting of fields — `college_code`, `college_name`, `address`
   - **Faculty**: consisting of fields — `college_code`, `faculty_code`, `faculty_name`, `qualification`, `experience` (in number of years), `department`, `address`. The field `college_code` is a foreign key referencing the `College` table.

   Generate queries for the following:

   (a) List all faculty members of a specified college whose experience is greater than or equal to 10 years.

   (b) List all faculty members of a specified college who have at least 10 years of experience but do not have an M.Tech degree.

   (c) List the faculty members of a specified college, department-wise, in non-decreasing order of their seniority.

   (d) List the names of colleges having more than a specified number of faculty members.

   (e) List the names of the colleges with the least and the most number of faculty members.

2. Create the following table for a Library Management System:

   - **Book**: consisting of fields — `accession_no`, `title`, `publisher`, `author`, `date_of_purchase`, `date_of_publishing`, `status`

   The `status` field can have one of the following values:

   - `issued`
   - `present in the Library`
   - `reference`
   - `cannot be issued`

   Based on the above table, write SQL queries for the following tasks:

   (a) List the total number of copies of each book in the library.

   (b) List the total number of reference copies for each book in the library.

   (c) For each book in the library, obtain:
   - the total number of issued copies,
   - the number of copies currently available in the library, and

- the number of reference copies.

(d) List the details of all books whose status is set to `"cannot be issued"` publisherwise.

(e) List the details of all books that are categorized as **new arrivals** (i.e., books purchased within the last 6 months).

(f) List the details of each **famous book**. A book is considered famous if:

- it was purchased within 1 year of its publishing date, and
- it has more than 10 total copies in the library.

3. **Create the following tables:**

- **Student** (`roll_no`, `name`, `date_of_birth`)
- **Course** (`course_id`, `name`, `fee`, `duration`)

Write SQL queries for the following:

(a) List the names of all students who are greater than 18 years of age and have opted for the B.Tech course.

(b) List the details of those courses whose fee is greater than that of the B.Tech course.

(c) List the details of the students who have opted for more than 2 courses.

(d) List the details (name, fee, and duration) of:

- the course that has been opted by the maximum number of students, and
- the course that has been opted by the least number of students.

(e) List the details of the student(s) who have opted every course.

4. **Create the following tables:**

- **Book** (`accession_no`, `title`, `publisher`, `year`, `date_of_purchase`, `status`)
- **Member** (`member_id`, `name`, `number_of_books_issued`, `max_limit`)
- **Books_Issue** (`accession_no`, `member_id`, `date_of_issue`)

Write SQL queries for the following:

(a) List all books that are due from students. A book is considered due if it was issued more than 15 days ago and has not yet been returned.

(b) List all members who cannot be issued any more books (i.e., their `number_of_books_issued` is equal to `max_limit`).

(c) List the details of:

- the book that has been issued to the maximum number of members, and
- the book that has been issued to the least number of members.

(d) List the details of:

- the book that has been issued to every member, and
- the book that has not yet been issued to any member.

5. **Create the following tables:**

- **Branch** (`branch_id`, `branch_name`, `branch_city`)
- **Customer** (`customer_id`, `customer_name`, `customer_city`)

- **Savings** (`customer_id`, `branch_id`, `saving_accno`, `balance`)
- **Loan** (`customer_id`, `branch_id`, `loan_accno`, `balance`)

Write SQL queries for the following:

(a) List the details of customers who live in the same city where they have an account.

(b) List the customers who have an account in a given branch city.

(c) List the customers who have accounts in more than one branch.

(d) List the details of customers who:

i. do not have a savings account but have a loan,

ii. do not have a loan but have a savings account,

iii. have both a loan and a savings account.

(e) List the names of customers who have no savings account at all but have loans in more than two branches.

(f) For each branch, produce a list showing:

- total number of customers,
- total number of customers with only a loan,
- total number of customers with only a savings account, and
- total number of customers with both loan and savings accounts.

(g) Find the details of the branch that has issued the maximum total amount of loan.

(h) Find the details of the branch that has not issued any loan at all.

(i) For each customer, produce a list showing the total savings balance and loan balance for all branches where they have either a loan, a savings account, or both.

6. A web based platform for a tourist resort that provides lodging on a booking basis has the following data storage.

- The details of the employees of the resort(Name, Address, Aadhar number, Mobile Number, Email ID, Joining Date, Salary).
- The details of the residents(Name, Address, Aadhar number, Gender, Age, Mobile Number, Email ID)
- The details of the companions(Name, Gender, Mobile Number, The details of the respective resident)
- The details of the rooms in the resort(Room Number, Room Type(A/C or Non A/C), Capacity(Single Bed or Double Bed))
- The Booking Details.
- The details of the Food items provided by the resort(Food Item, Type(Vegetarian or Non-Vegetarian, Price)
- The details of the food orders

The platform is supposed to provide the following functionalities.

(a) Print the details of the residents who has more than three companions in a single booking.

(b) Print the details of the residents along with the number of companions for a specified period.

(c) Print the details of the residents who reserved more than two A/C rooms in at least two different bookings.

(d) Print the details of the food item(s) ordered by maximum number of residents an the details of the food item(s) ordered b the minimum number of residents.

(e) Print the details of the food item(s) in the non-decreasing order of preference in a specific period. The preference is to be computed on the basis of two factors- the number of orders for that item in the period an the price for the item. The item with the highest number of orders and least price is considered to have most preference.

Create an ER diagram for this platform with the required entities and relationships. Convert the diagram to relational schema and write SQL queries for the creation of this database and information retrieval as per the questions given afore.

2 PL/SQL Experiments

1. Write a PL/SQL script to check whether a number is narcissistic or not. (Hint:- A d digit number n is said to be narcissistic if the sum of the sum of the d^{th} powers of the digits of n is equal to n itself. For example, 153 is a narcissistic number since $1^3 + 5^3 + 3^3 = 153$.)

2. Write a PL/SQL script to check whether a string (read from the user) is a plaindrome or not.

3. Write a PL/ SQL script that prints the first n fibonacci numbers.

4. Write a PL/SQL program that reads a positive integer n, forms two seperate tables namely *PRIME* and *COMPOSITE* containing the first n prime numbers and the first n composite numbers, print the contents of the two tables and then delete the two tables.

5. Write a PL/SQL program that reads a positive integer n, forms two seperate tables namely *SQUARE* and *CUBE* containing the perfect squares upto (including) n and the perfect cubes (including) n, print the contents of the two tables and then delete the two tables.

6. Given the schema Employee (empid, empname, joining date, relieving date, salary)

   (a) Find the service (in years) for each relieved employee.

   (b) Find the Pension amount to be paid to each relieved employee. (Pension is equal to the years of service *salary divided by 100.

   Use cursors.

7. The following table shows the details of the books available in a library.
   Book(Accno, Title, Author, Publisher, Eition, Number of copies)
   Write a trigger that displays the total number of tuples in the table on each insertion, deletion and updation.

8. The following table shows the salary information of employees in a company.
   Employee(empid, empname, salary)
   Write a trigger that causes insertion of a new entry into the table INCREMENT(empid, incr), if the difference arising due to an updation of the salary of an existing employee is greater than Rs. 1000/-.

9. The following table records the marks of various subjects for the B.Tech students of the CSE Department of RIT Kottayam.
   STUDENTMARK(StudentName, KTUID, Semester, Subject Name, Subject Code, Credit, Internal Mark, End Semster Exam Grade)
   Write a trigger that on each insertion/updation of the entries in the STUDENTMARK table leads to the insertion/updation of the SGPA of the respective student stored in the table SGPACSE(KTUID, Semester, SGPA) which in turn leads to the insertion/updation of the CGPA of the respective student stored in the table CGPACSE(KTUDID, CGPA).