Department of Computer Science and Engineering,
Anna University, Chennai- 600025
CS23401 – Database Management Systems (R 2023)
Practical
Jan- May 2023
Year/Sem/Batch: II/IV/ P

Exercise: 04	SQL Queries	08 – Jan - 2026

Observation (5 Marks)

1. Write the syntax with example for the following **JOIN operations**:
   a. INNER JOIN
   b. LEFT OUTER JOIN
   c. RIGHT OUTER JOIN
   d. FULL OUTER JOIN
2. Write the syntax and example for creating and dropping a **VIEW**.
3. Explain the following **constraints with syntax and example**:
   a. PRIMARY KEY
   b. FOREIGN KEY
   c. UNIQUE
   d. NOT NULL
4. What is **CASCADE operation**? Explain the following with examples:
   a. ON DELETE CASCADE
   b. ON UPDATE CASCADE
5. Write short notes on:
   a. CHECK Constraint
   b. DEFAULT Constraint

Execution (15 Marks)

(Create the required tables such as STUDENT, DEPARTMENT, COURSE, FACULTY, ENROLLMENT, MARKS with appropriate attributes and constraints).

1. Create a **DEPARTMENT** table with **PRIMARY KEY** and **UNIQUE** constraint.
2. Create a **STUDENT** table with the following constraints:
   a. **PRIMARY KEY** on Student_ID
   b. **NOT NULL** on Student_Name
   c. **CHECK** constraint on Age
   d. **DEFAULT** constraint on City
3. Create a **COURSE** table with **PRIMARY KEY** and **CHECK constraint** on Credits.
4. Create a **FACULTY** table with **PRIMARY KEY** and **UNIQUE constraint** on Email.
5. Create an **ENROLLMENT** table with **FOREIGN KEY** referencing STUDENT and COURSE tables using:
   a. ON DELETE CASCADE
   b. ON UPDATE CASCADE
6. Insert valid records into all tables and verify constraint enforcement.
7. Attempt to insert duplicate and NULL values to demonstrate **constraint violation**.
8. Display student details along with department name using **INNER JOIN**.
9. Display all students and their enrolled courses using **LEFT JOIN**.
10. Display all courses and enrolled students using **RIGHT JOIN**.

11. Display faculty and courses taught using **FULL OUTER JOIN**.
12. Create a VIEW named **Student_Course_View** showing student name, course name, and credits.
13. Display records from **Student_Course_View**.
14. Insert a record through the VIEW and observe its effect on the base table.
15. Demonstrate **ON DELETE CASCADE** by deleting a student record and showing automatic deletion in ENROLLMENT table.

Spot (5 Marks)

1. Write a SQL query to display **student name, department name, and course name** for students who are **not enrolled in any course**.
2. Write a SQL statement to **create a FOREIGN KEY** in the **ENROLLMENT** table referencing **STUDENT(Student_ID)** with both **ON DELETE CASCADE** and **ON UPDATE CASCADE**.
3. Write a SQL query to display the **count of students per department**, but show **only those departments** where the count is **greater than the average student count** across all departments.
4. Write a SQL query to display **student details** whose **enrollment records will be deleted** when a particular **course is removed**.
5. Write a SQL query using a **VIEW** to display students whose **course credits are greater than the average credits** of all courses.