**-- SQL DEVELOPER TASK - 2**

**-- Step 1: Create Required Tables**

**-- Courses Table use studentmanagement ;**

CREATE TABLE Courses ( course\_id INT AUTO\_INCREMENT PRIMARY KEY, course\_name VARCHAR(100), course\_description TEXT );

**-- Enrolments Table**

CREATE TABLE Enrolments ( enrolment\_id INT AUTO\_INCREMENT PRIMARY KEY, student\_id INT, course\_id INT, enrolment\_date DATE, FOREIGN KEY (student\_id) REFERENCES Students(StudentID), FOREIGN KEY (course\_id) REFERENCES Courses(course\_id) );

**-- Step 2: Insert Sample Data**

**-- Insert into Courses**

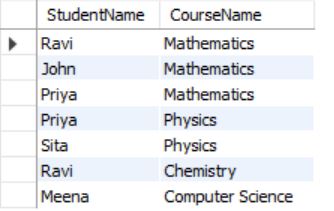
INSERT INTO Courses (course\_name, course\_description) VALUES ('Mathematics', 'Advanced mathematics course'), ('Physics', 'Basic physics course'), ('Chemistry', 'Organic and inorganic chemistry'), ('Computer Science', 'Intro to programming and databases');

**-- Insert into Enrolments**

INSERT INTO Enrolments (student\_id, course\_id, enrolment\_date) VALUES (1, 1, '2024-01-10'), (2, 2, '2024-01-12'), (1, 3, '2024-02-01'), (4, 2, '2024-03-15'), (5, 1, '2024-04-20'), (2, 1, '2024-05-10'), (6, 4, '2024-05-18');

**-- Task 1: List all students and the courses they are enrolled in**

SELECT s.Name AS StudentName, c.course\_name AS CourseName FROM Students s INNER JOIN Enrolments e ON s.StudentID = e.student\_id INNER JOIN Courses c ON e.course\_id = c.course\_id;

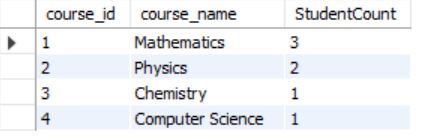


**Explanation:**

* Performs two INNER JOINs: Students → Enrolments → Courses.
* Returns student names along with their respective course names.

**-- Task 2:** **Find number of students enrolled in each course**

* SELECT c.course\_id, c.course\_name, COUNT(e.student\_id) AS StudentCount FROM Courses c LEFT JOIN Enrolments e ON c.course\_id = e.course\_id GROUP BY c.course\_id, c.course\_name;

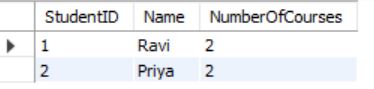


**Explanation:**

* LEFT JOIN ensures all courses are included, even with no students.
* GROUP BY groups data by each course.
* COUNT() gives the number of students per course.

**-- Task 3:** **List students who have enrolled in more than one course**

SELECT s.StudentID, s.Name, COUNT(e.course\_id) AS NumberOfCourses FROM Students s JOIN Enrolments e ON s.StudentID = e.student\_id GROUP BY s.StudentID, s.Name HAVING COUNT(e.course\_id) > 1;



**Explanation:**

* Groups students and counts their courses.
* HAVING COUNT > 1 filters only those enrolled in more than one course.

**-- Task 4: Find courses with no enrolled students**

SELECT c.course\_id, c.course\_name FROM Courses c LEFT JOIN Enrolments e ON c.course\_id = e.course\_id WHERE e.enrolment\_id IS NULL;



**Explanation:**

* LEFT JOIN brings in all courses.
* WHERE e.enrolment\_id IS NULL filters out the ones with no enrollments.