ECON485 – Homework Assignment 3

Course Registration System – SQL (Part II)

Part 1 – Basic JOIN Queries

Task 1 – List Students and Their Registered Sections

```
SELECT
  s.first_name || ' ' || s.last_name AS student_name,
  c.course_code,
  sec.section_number
FROM Enrollment e
JOIN Student s ON s.student_id = e.student_id
JOIN Section sec ON sec.section_id = e.section_id
JOIN Course c ON c.course_id = sec.course_id
ORDER BY student_name, c.course_code;
```

Task 2 – Show Courses with Total Student Counts

```
SELECT
  c.course_code,
  c.course_title,
  COUNT(e.student_id) AS total_students
FROM Course c
LEFT JOIN Section sec ON sec.course_id = c.course_id
LEFT JOIN Enrollment e ON e.section_id = sec.section_id
GROUP BY c.course_id, c.course_code, c.course_title
ORDER BY c.course_code;
```

Task 3 – List All Prerequisites for Each Course

```
SELECT
  c.course_code AS target_course,
  rc.course_code AS prerequisite_course,
  p.minimum_grade
FROM Prerequisite p
JOIN Course c  ON c.course_id = p.course_id
JOIN Course rc ON rc.course_id = p.required_course_id
ORDER BY c.course_code, rc.course_code;
```

PART 2 – Intermediate JOIN Query

Task 4 – Identify Students Eligible to Take a Course

```
SELECT
  s.first_name || ' ' || s.last_name AS student_name,
  c.course_code AS target_course,
  rc.course_code AS prerequisite_course,
  cc.letter_grade AS student_grade,
  p.minimum_grade,
  CASE
    WHEN cc.letter_grade IS NULL THEN 'NOT ELIGIBLE'
    WHEN
      (CASE
        WHEN cc.letter_grade = 'A' THEN 4
        WHEN cc.letter_grade = 'B' THEN 3
        WHEN cc.letter_grade = 'C' THEN 2
        WHEN cc.letter_grade = 'D' THEN 1
        ELSE 0
      END)
      >=
      (CASE
        WHEN p.minimum_grade = 'A' THEN 4
        WHEN p.minimum_grade = 'B' THEN 3
        WHEN p.minimum_grade = 'C' THEN 2
        WHEN p.minimum_grade = 'D' THEN 1
        ELSE 0
      END)
    THEN 'ELIGIBLE'
    ELSE 'NOT ELIGIBLE'
  END AS eligibility_status
FROM Prerequisite p
JOIN Course c ON c.course_id = p.course_id
JOIN Course rc ON rc.course_id = p.required_course_id
JOIN Student s
LEFT JOIN CompletedCourses cc
  ON cc.student_id = s.student_id
 AND cc.course_id  = p.required_course_id
LEFT JOIN Enrollment e
  ON e.student_id = s.student_id
 AND e.section_id IN (
   SELECT section_id FROM Section WHERE course_id = p.course_id
 )
WHERE e.student_id IS NULL
ORDER BY student_name, prerequisite_course;
```

PART 3 – Advanced JOIN Query

Task 5 – Detect Students Who Registered Without Meeting Prerequisites

```
SELECT
  s.first_name || ' ' || s.last_name AS student_name,
  c.course_code AS registered_course,
  rc.course_code AS prerequisite_course,
  cc.letter_grade AS student_grade,
  p.minimum_grade
FROM Enrollment e
JOIN Student s ON s.student_id = e.student_id
JOIN Section sec ON sec.section_id = e.section_id
JOIN Course c ON c.course_id = sec.course_id
JOIN Prerequisite p ON p.course_id = c.course_id
JOIN Course rc ON rc.course_id = p.required_course_id
LEFT JOIN CompletedCourses cc
  ON cc.student_id = s.student_id
 AND cc.course_id  = p.required_course_id
WHERE
  cc.letter_grade IS NULL
  OR
  (CASE
    WHEN cc.letter_grade = 'A' THEN 4
    WHEN cc.letter_grade = 'B' THEN 3
    WHEN cc.letter_grade = 'C' THEN 2
    WHEN cc.letter_grade = 'D' THEN 1
    ELSE 0
  END)
  <
  (CASE
    WHEN p.minimum_grade = 'A' THEN 4
    WHEN p.minimum_grade = 'B' THEN 3
    WHEN p.minimum_grade = 'C' THEN 2
    WHEN p.minimum_grade = 'D' THEN 1
    ELSE 0
  END)
ORDER BY student_name, registered_course;
```