

```

-- =====
-- 0. KHỞI TẠO CƠ SỞ DỮ LIỆU
-- =====
-- Xóa và tạo lại schema để script chạy nhiều lần không lỗi
DROP SCHEMA IF EXISTS edu_demo;
CREATE SCHEMA edu_demo;
USE edu_demo;

-- -----
-- BẢNG: students (sinh viên)
-- -----
CREATE TABLE students (
    id            INT PRIMARY KEY,
    name          VARCHAR(100) NOT NULL,
    gender        CHAR(1) CHECK (gender IN ('M','F')),
    birth_date    DATE
);

-- -----
-- BẢNG: courses (môn học)
-- -----
CREATE TABLE courses (
    id            INT PRIMARY KEY,
    title         VARCHAR(100) NOT NULL,
    credits       INT CHECK (credits BETWEEN 1 AND 6)
);

-- -----
-- BẢNG: enrollments (ĐK học - quan hệ nhiều-nhiều)
-- -----
CREATE TABLE enrollments (
    student_id    INT,
    course_id     INT,
    enroll_date   DATE,
    grade         CHAR(2),
    PRIMARY KEY(student_id, course_id),
    FOREIGN KEY (student_id) REFERENCES students(id),
    FOREIGN KEY (course_id) REFERENCES courses(id)
);

-- -----
-- BẢNG: instructors (giảng viên)
-- -----
CREATE TABLE instructors (
    id            INT PRIMARY KEY,
    name          VARCHAR(100),
    hire_date     DATE
);

-- -----
-- BẢNG: teachings (phân công giảng dạy)
-- -----
CREATE TABLE teachings (
    instructor_id INT,
    course_id     INT,
    semester      VARCHAR(10),      -- ví dụ '2024A'
    PRIMARY KEY(instructor_id, course_id, semester),
    FOREIGN KEY(instructor_id) REFERENCES instructors(id),
    FOREIGN KEY(course_id)      REFERENCES courses(id)
);

-- =====
-- 1. DỮ LIỆU MẪU
-- =====

```

```

INSERT INTO students (id, name, gender, birth_date) VALUES
(1, 'Nguyen Van A', 'M', '2001-05-17'),
(2, 'Tran Thi B', 'F', '2000-12-02'),
(3, 'Le Van C', 'M', '1999-04-21'),
(4, 'Pham Thi D', 'F', '2002-07-09');

INSERT INTO courses (id, title, credits) VALUES
(101, 'Hệ quản trị CSDL', 3),
(102, 'Thuật toán', 4),
(103, 'Hệ điều hành', 3),
(104, 'Khai phá dữ liệu', 3);

INSERT INTO enrollments (student_id, course_id, enroll_date, grade) VALUES
(1,101,'2024-01-10','A'),
(1,102,'2024-01-10','B+'),
(2,101,'2024-01-10','A-'),
(2,103,'2024-01-10','B'),
(3,104,'2024-01-10','B+'),
(4,101,'2024-01-10','A'),
(4,102,'2024-01-10','A');

INSERT INTO instructors (id, name, hire_date) VALUES
(11,'TS. Hoang','2018-09-01'),
(12,'TS. Pham','2020-02-15');

INSERT INTO teachings (instructor_id, course_id, semester) VALUES
(11,101,'2024A'),
(11,104,'2024A'),
(12,102,'2024A'),
(12,103,'2024A');

```

```

COMMIT;

```

```

-- =====
-- 2. BÀI TẬP & LỜI GIẢI
-- =====

/*****
BÀI TẬP 1
-----
Liệt kê tất cả sinh viên sinh sau năm 2000.
*****/

-- Lời giải:
-- SELECT * FROM students WHERE birth_date >= '2000-01-01';

/*****
BÀI TẬP 2
-----
Hiển thị số lượng môn học mà mỗi sinh viên đang đăng ký,
sắp xếp giảm dần theo số lượng.
*****/

-- Lời giải:
-- SELECT s.name,
--        COUNT(e.course_id) AS so_mon
-- FROM students s
-- JOIN enrollments e ON e.student_id = s.id
-- GROUP BY s.id, s.name
-- ORDER BY so_mon DESC;

/*****
BÀI TẬP 3
-----
Liệt kê tên các môn có số tín chỉ lớn hơn 2.
*****/

```

```

-- Lời giải:
-- SELECT title
--   FROM courses
--  WHERE credits > 2;

/*****
BÀI TẬP 4
-----
Với mỗi môn học, liệt kê giảng viên được tuyển trước năm 2019.
*****/
-- Lời giải:
-- SELECT c.title,
--        i.name AS giang_vien
--   FROM courses c
--  JOIN teachings t ON t.course_id = c.id
--  JOIN instructors i ON i.id = t.instructor_id
-- WHERE i.hire_date < '2019-01-01';

/*****
BÀI TẬP 5
-----
Tính điểm trung bình (GPA) theo thang 4 cho từng môn.
*****/
-- Gợi ý: Chuyển đổi điểm chữ sang điểm số.
-- Lời giải:
-- SELECT c.title,
--        AVG(
--          CASE grade
--            WHEN 'A' THEN 4.0
--            WHEN 'A-' THEN 3.7
--            WHEN 'B+' THEN 3.3
--            WHEN 'B' THEN 3.0
--            WHEN 'B-' THEN 2.7
--            WHEN 'C+' THEN 2.3
--            WHEN 'C' THEN 2.0
--          END
--        ) AS gpa_tb
--   FROM courses c
--  JOIN enrollments e ON e.course_id = c.id
-- GROUP BY c.id, c.title;

/*****
BÀI TẬP 6
-----
Liệt kê sinh viên chưa từng đăng ký môn 'Thuật toán'.
*****/
-- Lời giải:
-- SELECT s.*
--   FROM students s
--  WHERE NOT EXISTS (
--    SELECT 1
--      FROM enrollments e
--     JOIN courses c ON c.id = e.course_id
--    WHERE e.student_id = s.id
--          AND c.title = 'Thuật toán'
--  );

/*****
BÀI TẬP 7
-----
Liệt kê 3 sinh viên trẻ nhất (ngày sinh lớn nhất).
*****/
-- Lời giải:
-- SELECT *

```

```
-- FROM students
-- ORDER BY birth_date DESC
-- LIMIT 3;
```

```

/*****
BÀI TẬP 8
-----

```

Hiển thị mỗi giảng viên cùng tổng số sinh viên họ dạy
(qua các môn được phân công).

```

*****/
-- Lời giải:
-- SELECT i.name,
--        COUNT(DISTINCT e.student_id) AS tong_sv
-- FROM instructors i
-- JOIN teachings t ON t.instructor_id = i.id
-- JOIN enrollments e ON e.course_id = t.course_id
-- GROUP BY i.id, i.name
-- ORDER BY tong_sv DESC;
```

```

/*****
BÀI TẬP 9
-----

```

Tạo VIEW tên v_student_courses gồm tên SV, tên môn, và điểm.

```

*****/
-- Lời giải:
-- CREATE OR REPLACE VIEW v_student_courses AS
-- SELECT s.name AS sinh_vien,
--        c.title AS mon_hoc,
--        e.grade
-- FROM students s
-- JOIN enrollments e ON e.student_id = s.id
-- JOIN courses c ON c.id = e.course_id;
```

```

/*****
BÀI TẬP 10
-----

```

Tạo chỉ mục trên enroll_date trong bảng enrollments
để tăng tốc truy vấn theo khoảng thời gian.

```

*****/
-- Lời giải (MySQL):
-- CREATE INDEX idx_enroll_date ON enrollments(enroll_date);
```

```

/*****
BÀI TẬP 11
-----

```

Sử dụng window function để xếp hạng sinh viên theo GPA.

```

*****/
-- Lời giải (MySQL 8):
-- WITH gpa AS (
--   SELECT s.id,
--          s.name,
--          AVG(
--            CASE grade
--              WHEN 'A' THEN 4.0
--              WHEN 'A-' THEN 3.7
--              WHEN 'B+' THEN 3.3
--              WHEN 'B' THEN 3.0
--              WHEN 'B-' THEN 2.7
--              WHEN 'C+' THEN 2.3
--              WHEN 'C' THEN 2.0
--            END
--          ) AS gpa_val
-- FROM students s
-- JOIN enrollments e ON e.student_id = s.id

```

```

-- GROUP BY s.id, s.name
--)
-- SELECT *,
--     RANK() OVER (ORDER BY gpa_val DESC) AS thu_hang
-- FROM gpa;

/*****
BÀI TẬP 12
-----
Trả về những sinh viên có điểm môn 'Hệ quản trị CSDL' cao hơn
điểm trung bình của chính môn đó.
*****/

-- Lời giải:
-- WITH avg_course AS (
--     SELECT AVG(
--         CASE grade
--         WHEN 'A' THEN 4.0
--         WHEN 'A-' THEN 3.7
--         WHEN 'B+' THEN 3.3
--         WHEN 'B' THEN 3.0
--         WHEN 'B-' THEN 2.7
--         WHEN 'C+' THEN 2.3
--         WHEN 'C' THEN 2.0
--         END
--     ) AS gpa_tb
-- FROM enrollments e
-- JOIN courses c ON c.id = e.course_id
-- WHERE c.title = 'Hệ quản trị CSDL'
--)
-- SELECT s.name,
--     e.grade
-- FROM students s
-- JOIN enrollments e ON e.student_id = s.id
-- JOIN courses c ON c.id = e.course_id
-- WHERE c.title = 'Hệ quản trị CSDL'
-- AND (
--     CASE e.grade
--     WHEN 'A' THEN 4.0
--     WHEN 'A-' THEN 3.7
--     WHEN 'B+' THEN 3.3
--     WHEN 'B' THEN 3.0
--     WHEN 'B-' THEN 2.7
--     WHEN 'C+' THEN 2.3
--     WHEN 'C' THEN 2.0
--     END
-- ) > (SELECT gpa_tb FROM avg_course);

-- =====
-- KẾT THÚC SCRIPT
-- =====

```