

**ĐẠI HỌC QUỐC GIA HÀ NỘI  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**



**BÁO CÁO**  
**BÀI TẬP LỚN CƠ SỞ DỮ LIỆU**

CHỦ ĐỀ: QUẢN LÝ GIÁO VỤ

NHÓM: 20

# DATABASE FINAL PROJECT REPORT

## INT2211 ASSIGNMENT

### GROUP 20 – K68-A-AI

## GIỚI THIỆU

### 1. Thành viên nhóm

HỌ VÀ TÊN	MÃ SINH VIÊN – LỚP	VAI TRÒ
NGUYỄN HOÀNG TÚ	23020428 – K68-A-AI2	Nhóm trưởng
ĐÀM LÊ MINH QUÂN	23020416 – K68-A-AI2	Thành viên
HOÀNG SỸ TOÀN	23020426 – K68-A-AI2	Thành viên
NGUYỄN KHÁNH TÙNG	23020434 – K68-A-AI2	Thành viên

### 2. GV hướng dẫn

Giảng viên: TS. Trần Hồng Việt

### 3. Tổng quan dự án

- Dự án được thiết kế phù hợp với khả năng quản lý hệ thống dữ liệu của một tổ chức giáo dục trong thực tế. Thông qua dự án chúng ta có thể có được góc nhìn sơ lược về nhiệm vụ quản lý và khai thác dữ liệu, đồng thời hình thành tư duy xây dựng mô hình quản lý dữ liệu.

## CHI TIẾT DỰ ÁN

### I. Phát biểu bài toán

- Trong thực tế, các cơ sở giáo dục từ cấp bậc nhỏ tới lớn đứng trước một yêu cầu rất lớn về vấn đề quản lý dữ liệu. Ví dụ, một cơ sở giáo dục bậc đại học có rất nhiều sinh viên, giáo viên và vô số những nhân tố thành phần khác, do đó việc phải đảm bảo tính minh bạch, rõ ràng của dữ liệu là vô cùng quan trọng để tránh những thiếu sót trong vận hành hoạt động của cơ sở giáo dục đó.

### II. Nghiệp vụ của hệ thống

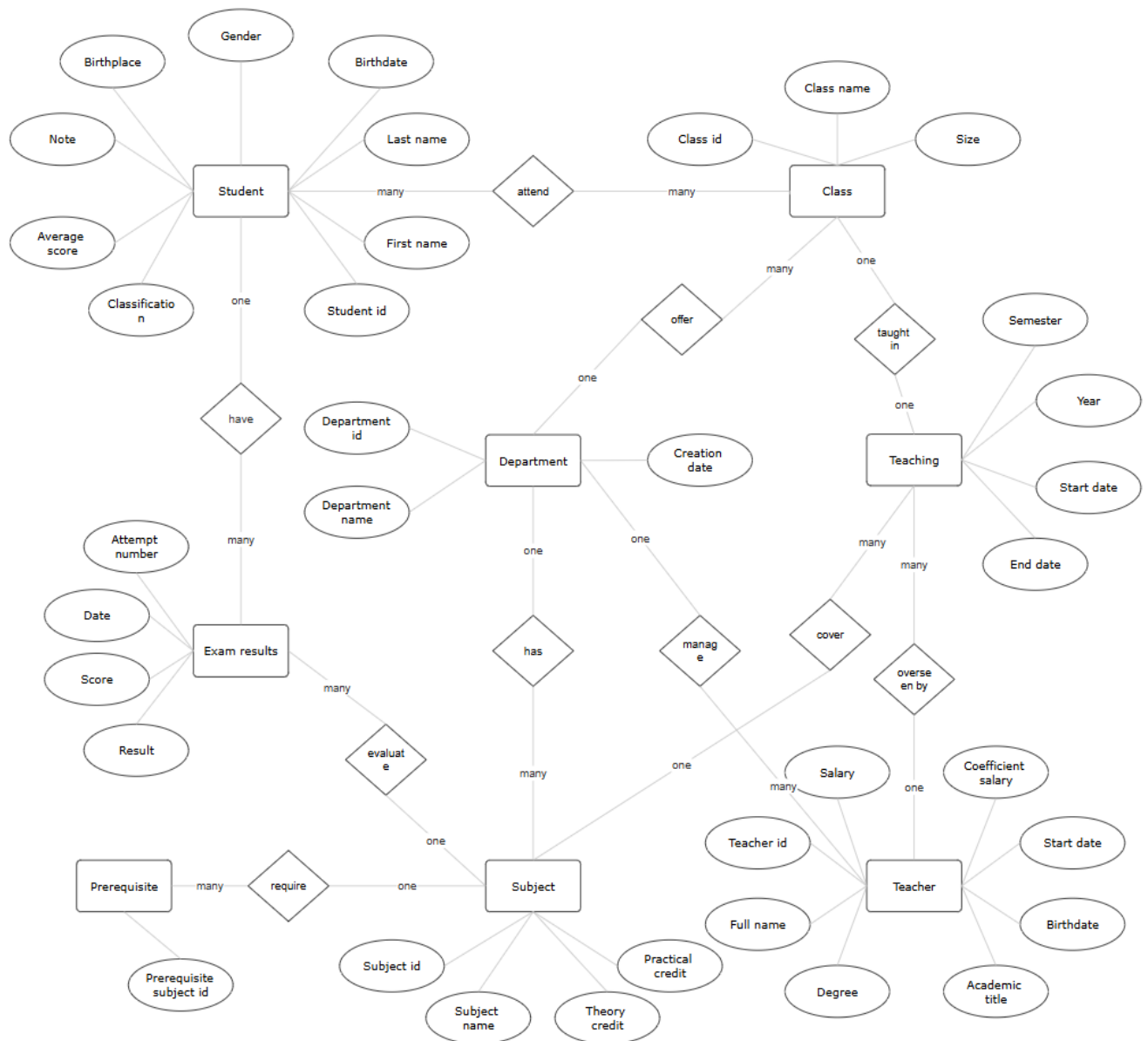
Nhằm giải quyết những nhu cầu thực tiễn đó, dự án sẽ tập trung vào phát triển một hệ thống quản lý giáo vụ, với các chức năng chính :

- **Quản lý thông tin sinh viên (student)** : Lưu trữ và quản lý chỉnh sửa, truy cập một số dữ liệu của sinh viên để theo dõi tiến độ học tập của sinh viên. Các thông tin cơ bản ở đây bao gồm họ tên, mã sinh viên, giới tính, ngày sinh, nơi sinh, điểm thi các môn, thông tin lớp học, khoa của sinh viên.
- **Quản lý thông tin giáo viên (teacher)** : Lưu trữ và quản lý chỉnh sửa, truy cập dữ liệu các giáo viên trong trường. Các thông tin cơ bản ở đây bao gồm họ tên, mã giáo viên, giới tính, ngày sinh, chức danh, trình độ học vấn, ngày bắt đầu công tác, thông tin về mức lương và khoa/viện mà giáo viên đó trực thuộc, thông tin về các lớp phụ trách và thời gian dạy.
- **Quản lý thông tin các khoa/viện trong trường (department)** : Lưu trữ và quản lý việc chỉnh sửa, truy cập dữ liệu của các khoa viện trực thuộc trường. Các thông tin cơ bản sẽ gồm mã khoa/viện, tên khoa/viện, ngày thành lập khoa/viện, mã trưởng khoa/viện.
- **Quản lý thông tin các lớp học (class)** : Lưu trữ và quản lý chỉnh sửa, khai thác dữ liệu của các lớp học trong trường để phục vụ cho việc phân bổ sinh viên và giáo viên. Các thông tin cơ bản bao gồm mã lớp học, tên lớp học, mã sinh viên của lớp trưởng, sĩ số lớp, mã giáo viên của giáo viên phụ trách lớp.
- **Quản lý thông tin các môn học (subject)** : Lưu trữ và quản lý chỉnh sửa, truy cập dữ liệu của các môn học trong chương trình đào tạo của trường, ghi nhận mối quan hệ tiền đề giữa các môn học nhằm đảm bảo sinh viên phải hoàn thành các môn học cơ bản trước khi đến với các môn học nâng cao hơn. Các môn học có dữ liệu chính bao gồm mã môn học, tên môn học, số tín chỉ lý thuyết, số tín chỉ thực hành, thông tin khoa/viện của môn học đó.

### III. Mô hình ER của hệ thống

- Mô hình ER của một hệ thống quản lý giáo vụ cơ bản có các thực thể và mối quan hệ được mô tả chi tiết trong sơ đồ sau :

Entity Relationship Diagram - Academic Administration Model



### IV. Mô hình quan hệ chuyển từ mô hình ER

• Sử dụng các câu lệnh SQL để thiết lập một cơ sở dữ liệu cho hệ thống được lưu trong MySQL Workbench

+ Tạo một database mới mang tên academic\_administration :

- 1 • `CREATE DATABASE IF NOT EXISTS academic_administration;`
- 2 • `USE academic_administration;`

+ Tạo bảng lưu các thuộc tính cho thực thể sinh viên :

```
-- Table: Học viên
• CREATE TABLE IF NOT EXISTS student (
    student_id CHAR(5),
    last_name VARCHAR(40),
    first_name VARCHAR(10),
    birth_date DATE,
    gender VARCHAR(3),
    birth_place VARCHAR(40),
    class_id CHAR(3),
    note VARCHAR(100),
    average_score DECIMAL(4,2),
    classification VARCHAR(10),
    PRIMARY KEY (student_id)
);
```

+ Tạo bảng lưu các thuộc tính cho giáo viên :

```
-- Table: Giáo viên
• CREATE TABLE IF NOT EXISTS teacher (
    teacher_id CHAR(4),
    full_name VARCHAR(40),
    degree VARCHAR(10),
    academic_title VARCHAR(10),
    gender VARCHAR(3),
    birth_date DATE,
    start_date DATE,
    coefficient_salary NUMERIC(4, 2),
    salary DECIMAL(15,2),
    department_id VARCHAR(4),
    PRIMARY KEY (teacher_id)
);
```

+ Tạo bảng lưu thực thể lớp học :

```
-- Table: Lớp
• CREATE TABLE IF NOT EXISTS class (
    class_id CHAR(3),
    class_name VARCHAR(40),
    class_leader_id CHAR(5),
    size TINYINT,
    head_teacher_id CHAR(4),
    PRIMARY KEY (class_id)
);
```

+ Tạo bảng lưu thực thể môn học :

```
-- Table: Môn học
• CREATE TABLE IF NOT EXISTS subject (
    subject_id VARCHAR(10),
    subject_name VARCHAR(40),
    theory_credit TINYINT,
    practical_credit TINYINT,
    department_id VARCHAR(4),
    PRIMARY KEY (subject_id)
);
```

+ Tạo bảng cho thực thể khoa :

```
-- Table: Khoa
• CREATE TABLE IF NOT EXISTS department (
    department_id VARCHAR(4),
    department_name VARCHAR(40),
    creation_date DATE,
    head_teacher_id CHAR(4),
    PRIMARY KEY (department_id)
);
```

+ Tạo bảng lưu dữ liệu giảng dạy :

```
-- Table: Giảng dạy
• CREATE TABLE IF NOT EXISTS teaching (
    class_id CHAR(3),
    subject_id VARCHAR(10),
    teacher_id CHAR(4),
    semester TINYINT,
    year SMALLINT,
    start_date DATE,
    end_date DATE,
    PRIMARY KEY (class_id, subject_id)
);
```

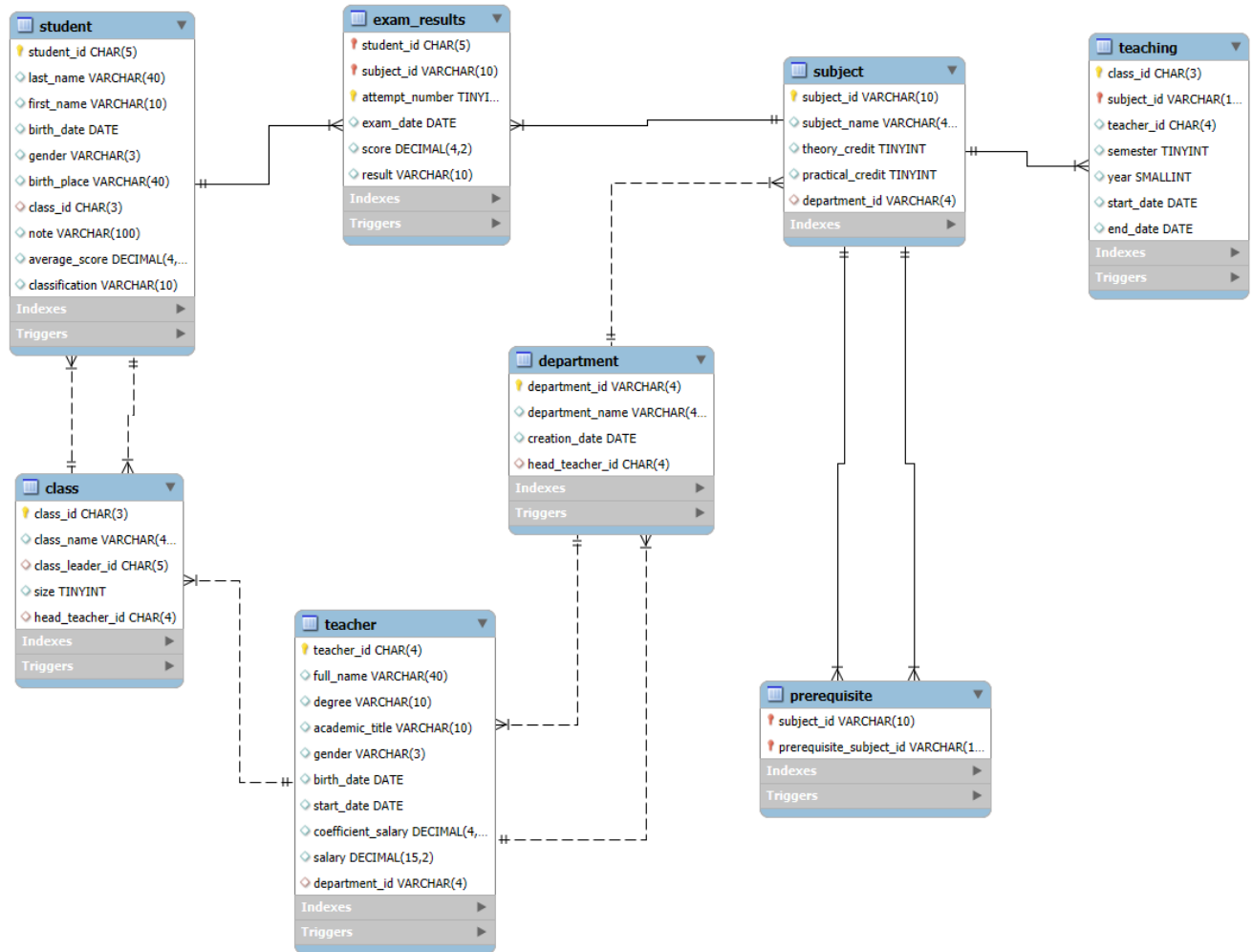
+ Tạo bảng lưu thông tin các kết quả thi :

```
-- Table: Kết quả thi
• CREATE TABLE IF NOT EXISTS exam_results (
    student_id CHAR(5),
    subject_id VARCHAR(10),
    attempt_number TINYINT,
    exam_date DATE,
    score NUMERIC(4, 2),
    result VARCHAR(10),
    PRIMARY KEY (student_id, subject_id, attempt_number)
);
```

+ Tạo bảng cho mục môn học tiền đề bắt buộc :

```
-- Table: Điều kiện
• CREATE TABLE IF NOT EXISTS prerequisite (
    subject_id VARCHAR(10),
    prerequisite_subject_id VARCHAR(10),
    PRIMARY KEY (subject_id, prerequisite_subject_id)
);
```

• Biểu diễn sơ đồ mô hình quan hệ :



**THÔNG TIN CÁC BẢNG ( Các thuộc tính được in đậm và gạch chân đóng vai trò là khóa chính trong bảng ) :**

1. **student** (**student\_id**, last\_name, first\_name, birth\_date, gender, birth\_place, class\_id, note, average\_score, classification)
2. **teacher** (**teacher\_id**, full\_name, degree, academic\_title, gender, birth\_date, start\_date, coefficient\_salary, salary, department\_id)
3. **class** (**class\_id**, class\_name, class\_leader\_id, size, head\_teacher\_id)
4. **department** (**department\_id**, department\_name, creation\_date, head\_teacher\_id)
5. **subject** (**subject\_id**, subject\_name, theory\_credit, practical\_credit, department\_id)



6. **teaching** (class\_id, subject\_id, teacher\_id, semester, year, start\_date, end\_date)

7. **prerequisite** (subject\_id, prerequisite\_subject\_id)

8. **exam\_results** (student\_id, subject\_id, attempt\_number, exam\_date, score, result)

#### *V. Từ điển dữ liệu*

Thuộc tính	Mô tả	Kiểu dữ liệu	Dung lượng tối đa
<b>student_id</b>	Mã sinh viên	CHAR	5
<b>last_name</b>	Tên họ của cá nhân	VARCHAR	40
<b>first_name</b>	Tên đầu của cá nhân	VARCHAR	10
<b>birth_date</b>	Ngày sinh của cá nhân	DATE	
<b>gender</b>	Giới tính cá nhân	VARCHAR	3
<b>birth_place</b>	Nơi sinh	VARCHAR	40
<b>class_id</b>	Mã lớp học	CHAR	3
<b>note</b>	Ghi chú về cá nhân	VARCHAR	100
<b>average_score</b>	Điểm trung bình	DECIMAL	(4,2)
<b>classification</b>	Dữ liệu phân loại sinh viên	VARCHAR	10
<b>subject_id</b>	Mã môn học	VARCHAR	10
<b>teacher_id</b>	Mã giảng viên	CHAR	4
<b>semester</b>	Học kỳ	TINYINT	
<b>year</b>	Năm học	SMALLINT	
<b>start_date</b>	Ngày bắt đầu giảng dạy	DATE	
<b>end_date</b>	Ngày kết thúc giảng dạy	DATE	
<b>department_id</b>	Mã khoa	VARCHAR	4
<b>department_name</b>	Tên khoa	VARCHAR	40

<b>creation_date</b>	Ngày thành lập khoa	DATE	
<b>head_teacher_id</b>	Mã giảng viên phụ trách lớp	CHAR	4
<b>subject_name</b>	Tên môn học	VARCHAR	40
<b>theory_credit</b>	Số tín chỉ lý thuyết	TINYINT	
<b>practical_credit</b>	Số tín chỉ thực hành	TINYINT	
<b>prerequisite_subject_id</b>	Mã môn học tiền đề	VARCHAR	10
<b>degree</b>	Học vị	VARCHAR	10
<b>academic_title</b>	Chức danh khoa học	VARCHAR	10
<b>coefficient_salary</b>	Hệ số lương	NUMERIC	(4, 2)
<b>salary</b>	Lương	DECIMAL	(15,2)
<b>class_name</b>	Tên lớp học	VARCHAR	40
<b>class_leader_id</b>	Mã sinh viên của lớp trưởng	CHAR	5
<b>size</b>	Sĩ số lớp	TINYINT	
<b>attempt_number</b>	Số lần thi	TINYINT	
<b>exam_date</b>	Ngày thi	DATE	
<b>score</b>	Điểm thi	NUMERIC	(4, 2)
<b>result</b>	Kết quả thi	VARCHAR	10

## VI. Các quan hệ và ràng buộc dữ liệu

### 1. Thêm các khóa ngoại và ràng buộc đối với một số thuộc tính

\* Thêm khóa ngoại

- `ALTER TABLE class`  
`ADD CONSTRAINT FK_class_leader FOREIGN KEY (class_leader_id) REFERENCES student(student_id),`  
`ADD CONSTRAINT FK_head_teacher FOREIGN KEY (head_teacher_id) REFERENCES teacher(teacher_id);`
- `ALTER TABLE student`  
`ADD CONSTRAINT FK_class_id FOREIGN KEY (class_id) REFERENCES class(class_id);`
- `ALTER TABLE teacher`  
`ADD CONSTRAINT FK_department_id FOREIGN KEY (department_id) REFERENCES department(department_id);`

- `ALTER TABLE teaching`  
`ADD CONSTRAINT FK_subject_id FOREIGN KEY (subject_id) REFERENCES subject(subject_id);`
- `ALTER TABLE subject`  
`ADD CONSTRAINT FK_department_id2 FOREIGN KEY (department_id) REFERENCES department(department_id);`
- `ALTER TABLE department`  
`ADD CONSTRAINT FK_head_teacher_department FOREIGN KEY (head_teacher_id) REFERENCES teacher(teacher_id);`
- `ALTER TABLE prerequisite`  
`ADD CONSTRAINT FK_subject_prerequisite FOREIGN KEY (subject_id) REFERENCES subject(subject_id),`  
`ADD CONSTRAINT FK_prerequisite_subject FOREIGN KEY (prerequisite_subject_id) REFERENCES subject(subject_id);`
- `ALTER TABLE exam_results`  
`ADD CONSTRAINT FK_student_id FOREIGN KEY (student_id) REFERENCES student(student_id),`  
`ADD CONSTRAINT FK_subject_exam FOREIGN KEY (subject_id) REFERENCES subject(subject_id);`

### \*Thêm các ràng buộc

-- Mã học viên có định dạng 5 ký tự: 3 ký tự đầu là mã lớp, 2 ký tự cuối là số

```
ALTER TABLE student
ADD CONSTRAINT CK_student_id_length CHECK (CHAR_LENGTH(student_id) = 5);
```

-- Giới tính chỉ có thể là nam hoặc nữ

```
ALTER TABLE student
ADD CONSTRAINT CK_gender_student CHECK (gender IN ('Nam', 'Nu'));
```

`ALTER TABLE teacher`

```
ADD CONSTRAINT CK_gender_teacher CHECK (gender IN ('Nam', 'Nu'));
```

-- Điểm thi có giá trị từ 0 đến 10

```
ALTER TABLE exam_results
ADD CONSTRAINT CK_exam_score CHECK (score BETWEEN 0 AND 10);
```

-- Kết quả thi phụ thuộc vào điểm

```
ALTER TABLE exam_results
ADD CONSTRAINT CK_exam_result CHECK (
    (score < 5 AND result = 'Khong dat') OR (score BETWEEN 5 AND 10 AND result = 'Dat')
);
```

```

-- Số lần thi không vượt quá 3 lần
ALTER TABLE exam_results
    ADD CONSTRAINT CK_attempts CHECK (attempt_number <= 3);

-- Học kỳ chỉ nhận các giá trị 1, 2, 3
ALTER TABLE teaching
    ADD CONSTRAINT CK_semester CHECK (semester IN (1, 2, 3));

-- Học vị giáo viên chỉ nhận giá trị cố định
ALTER TABLE teacher
    ADD CONSTRAINT CK_academic_degree CHECK (degree IN ('CN', 'KS', 'Ths', 'TS', 'PTS'));

-- Học viên ít nhất là 18 tuổi
ALTER TABLE student ADD CONSTRAINT CK_student_age
CHECK (2024 - YEAR(birth_date) >= 18);

-- Giảng dạy một môn học ngày bắt đầu (start_date) phải nhỏ hơn ngày kết thúc (end_date)
ALTER TABLE teaching ADD CONSTRAINT CK_teaching_dates
CHECK (start_date < end_date);

-- Giáo viên khi vào làm ít nhất là 22 tuổi
ALTER TABLE teacher ADD CONSTRAINT CK_teacher_age
CHECK (YEAR(start_date) - YEAR(birth_date) >= 22);

-- Tất cả các môn học đều có số tín chỉ lý thuyết và tín chỉ thực hành chênh lệch nhau không quá 3
ALTER TABLE subject ADD CONSTRAINT CK_credit_diff
CHECK (ABS(theory_credit - practical_credit) <= 3 OR practical_credit = 0);

```

## Phân tích các mối quan hệ và ràng buộc

- **student** và **class** : có mối quan hệ thông qua trường class\_id, trường class\_leader\_id và student\_id, trong đó class\_leader\_id tham chiếu đến student\_id -> Mỗi lớp chỉ có 1 lớp trưởng, 1 lớp có thể có nhiều sinh viên, 1 sinh viên có thể học ở nhiều lớp.
- **student** và **exam\_results** : liên kết qua trường student\_id -> Mỗi một bản ghi lưu kết quả thi chỉ thuộc về một sinh viên duy nhất, một học sinh có thể có nhiều kết quả thi.
- **teacher** và **class** : liên kết qua trường head\_teacher\_id của bảng class tham chiếu đến trường teacher\_id của bảng teacher -> Một lớp chỉ có một giáo viên đứng đầu, một giáo viên chỉ có thể phụ trách một lớp

- **teacher** và **department** : liên kết qua trường department\_id, trường head\_teacher\_id tham chiếu đến trường teacher\_id -> Mỗi khoa được quản lý bởi một giáo viên trưởng khoa, một khoa có thể có nhiều giáo viên trực thuộc.
- **subject** và **exam\_results** : liên kết thông qua trường subject\_id -> Mỗi một kết quả thi chỉ liên hệ với một môn học, một môn học thì có thể có nhiều kết quả thi.
- **subject** và **prerequisite** : liên kết thông qua việc các trường subject\_id và prerequisite\_subject\_id của bảng **prerequisite** tham chiếu đến trường subject\_id của bảng **subject** -> Một môn học có thể yêu cầu một hoặc nhiều môn tiền đề bắt buộc.
- **subject** và **department** : liên kết thông qua trường department\_id -> Một môn học chỉ thuộc về một khoa thông qua trường department\_id, một khoa có thể có nhiều môn học.
- **subject** và **teaching** : có mối liên kết thông qua trường subject\_id -> Một giáo viên sẽ dạy một môn học trong một lớp vào một học kỳ và năm học cụ thể
- **Các ràng buộc khác :**
  - + Đảm bảo đúng định dạng mã sinh viên (5 kí tự, 3 kí tự đầu là mã lớp, 2 kí tự sau là số)
  - + Thông tin giới tính chỉ có thể là nam hoặc nữ
  - + Điểm thi mang giá trị phù hợp (từ 0-10)
  - + Kết quả thi (result) phụ thuộc vào điểm thi (score)
  - + Số lần thi (attempt\_number) không vượt quá 3 lần
  - + Dữ liệu học kỳ (semester) chỉ có thể là 1,2 hoặc 3
  - + Học vị giảng viên chỉ có thể nhận một trong các dữ liệu (CN, KS, Ths, TS, PTS)
  - + Tuổi sinh viên từ 18 trở lên

- + Ngày bắt đầu giảng dạy phải nhỏ hơn ngày kết thúc giảng dạy
- + Độ tuổi tối thiểu của giảng viên là 22
- + Tất cả các môn học đều có số tín chỉ lý thuyết và tín chỉ thực hành chênh lệch nhau không quá 3.

## ***VII. Thêm dữ liệu phù hợp cho cơ sở dữ liệu quản lý giáo vụ***

\*Sử dụng các câu lệnh INSERT INTO ....VALUES (...):

```
USE academic_administration;

-- Disable foreign key checks
SET FOREIGN_KEY_CHECKS = 0;
SET SQL_SAFE_UPDATES = 0;

-- Insert data into the DEPARTMENT table
INSERT INTO department (department_id, department_name, creation_date, head_teacher_id)
VALUES
('KHMT', 'Khoa học máy tính', '2005-06-07', 'GV01'),
('HTTT', 'Hệ thống thông tin', '2005-06-07', 'GV02'),
('CNPM', 'Công nghệ phần mềm', '2005-06-07', 'GV04'),
('MTT', 'Mạng và truyền thông', '2005-10-20', 'GV03'),
('KTMT', 'Kỹ thuật máy tính', '2005-12-20', null);

-- Insert data into the CLASS table
INSERT INTO class (class_id, class_name, class_leader_id, size, head_teacher_id)
VALUES
('K11', 'Lớp 1 khoa 1', 'K1108', 11, 'GV07'),
('K12', 'Lớp 2 khoa 1', 'K1205', 12, 'GV09'),
('K13', 'Lớp 3 khoa 1', 'K1305', 12, 'GV14');

-- Insert data into the SUBJECT table
INSERT INTO subject (subject_id, subject_name, theory_credit, practical_credit, department_id)
VALUES
('THDC', 'Tin học đại cương', 4, 1, 'KHMT'),
('CTRR', 'Cấu trúc rời rạc', 5, 0, 'KHMT'),
('CSDL', 'Cơ sở dữ liệu', 3, 1, 'HTTT'),
('CTDLGT', 'Cấu trúc dữ liệu và giải thuật', 3, 1, 'KHMT'),
('PTTKTT', 'Phân tích thiết kế thuật toán', 3, 0, 'KHMT'),
('DHMT', 'Đồ họa máy tính', 3, 1, 'KHMT'),
('KTMT', 'Kiến trúc máy tính', 3, 0, 'KTMT'),
('TKCSDL', 'Thiết kế cơ sở dữ liệu', 3, 1, 'HTTT'),
('PTTKHTTT', 'Phân tích thiết kế hệ thống thông tin', 4, 1, 'HTTT'),
('HDH', 'Hệ điều hành', 4, 0, 'KTMT'),
('NMCNPM', 'Nhập môn công nghệ phần mềm', 3, 0, 'CNPM'),
('LTCFW', 'Lập trình C for win', 3, 1, 'CNPM'),
('LTHDT', 'Lập trình hướng đối tượng', 3, 1, 'CNPM');
```

-- Insert data into the PREREQUISITE table

```
INSERT INTO prerequisite (subject_id, prerequisite_subject_id)
VALUES
('CSDL', 'CTRR'),
('CTDLGT', 'CSDL'),
('CTDLGT', 'THDC'),
('PTTKTT', 'THDC'),
('PTTKTT', 'CTDLGT'),
('DHMT', 'THDC'),
('LTHDT', 'THDC'),
('PTTKHTTT', 'CSDL');
```

-- Insert data into the TEACHER table

```
INSERT INTO teacher (teacher_id, full_name, degree, academic_title, gender,
birth_date, start_date, coefficient_salary, salary, department_id)
VALUES
('GV01', 'Ho Thanh Son', 'PTS', 'GS', 'Nam', '1950-02-05', '2004-11-01', 5, 2250000, 'KHMT'),
('GV02', 'Tran Tam Thanh', 'TS', 'PGS', 'Nam', '1965-12-17', '2004-04-20', 4.5, 2025000, 'KHMT'),
('GV03', 'Do Nghiem Phung', 'TS', 'GS', 'Nu', '1950-01-08', '2004-09-23', 4, 1800000, 'CNPM'),
('GV04', 'Tran Nam Son', 'TS', 'PGS', 'Nam', '1961-02-22', '2005-12-01', 4.5, 2025000, 'KTMT'),
('GV05', 'Mai Thanh Danh', 'ThS', 'GV', 'Nam', '1958-12-03', '2005-12-01', 3, 1350000, 'HTTT'),
('GV06', 'Tran Doan Hung', 'TS', 'GV', 'Nam', '1953-11-03', '2005-12-01', 4.5, 2025000, 'KHMT'),
('GV07', 'Nguyen Minh Tien', 'ThS', 'GV', 'Nam', '1971-11-23', '2005-01-03', 4, 1800000, 'KHMT'),
('GV08', 'Le Thi Tran', 'KS', NULL, 'Nu', '1974-03-26', '2005-01-03', 1.69, 760500, 'KHMT'),
('GV09', 'Nguyen To Lan', 'ThS', 'GV', 'Nu', '1966-12-31', '2005-01-03', 4, 1800000, 'HTTT'),
('GV10', 'Le Tran Anh Loan', 'KS', NULL, 'Nu', '1972-07-17', '2005-01-03', 1.86, 837000, 'CNPM'),
('GV11', 'Ho Thanh Tung', 'CN', 'GV', 'Nam', '1980-12-01', '2005-05-15', 2.67, 1201500, 'MTT'),
('GV12', 'Tran Van Anh', 'CN', NULL, 'Nu', '1981-03-29', '2005-05-15', 1.69, 760500, 'CNPM'),
('GV13', 'Nguyen Linh Dan', 'CN', NULL, 'Nu', '1980-05-23', '2005-05-15', 1.69, 760500, 'KTMT'),
('GV14', 'Truong Minh Chau', 'ThS', 'GV', 'Nu', '1976-11-30', '2005-05-15', 3, 1350000, 'MTT'),
('GV15', 'Le Ha Thanh', 'ThS', 'GV', 'Nam', '1978-04-05', '2005-05-15', 3, 1350000, 'KHMT');
```

-- Insert data into the TEACHING table

```
INSERT INTO teaching (class_id, subject_id, teacher_id, semester, year, start_date, end_date)
VALUES
('K11', 'THDC', 'GV07', 1, 2006, '2006-01-02', '2006-05-12'),
('K12', 'THDC', 'GV06', 1, 2006, '2006-01-02', '2006-05-12'),
('K13', 'THDC', 'GV15', 1, 2006, '2006-01-02', '2006-05-12'),
('K13', 'CTDLGT', 'GV15', 2, 2006, '2006-06-01', '2006-07-15'),
('K11', 'CTDLGT', 'GV15', 3, 2006, '2006-08-01', '2006-12-15'),
('K12', 'CTDLGT', 'GV15', 3, 2006, '2006-08-01', '2006-12-15'),
('K11', 'CTRR', 'GV02', 1, 2006, '2006-01-09', '2006-05-17'),
('K12', 'CTRR', 'GV02', 1, 2006, '2006-01-09', '2006-05-17'),
('K13', 'CTRR', 'GV08', 1, 2006, '2006-01-09', '2006-05-17'),
('K11', 'CSDL', 'GV05', 2, 2006, '2006-06-01', '2006-07-15'),
('K12', 'CSDL', 'GV09', 2, 2006, '2006-06-01', '2006-07-15'),
('K13', 'CSDL', 'GV05', 3, 2006, '2006-08-01', '2006-12-15'),
('K13', 'DHMT', 'GV07', 3, 2006, '2006-08-01', '2006-12-15'),
('K11', 'HDH', 'GV04', 1, 2007, '2007-01-02', '2007-02-18'),
('K12', 'HDH', 'GV04', 1, 2007, '2007-01-02', '2007-03-20'),
('K11', 'DHMT', 'GV07', 1, 2007, '2007-02-18', '2007-03-20');
```



```
-- insert data into STUDENT table
```

```
INSERT INTO student (student_id, last_name, first_name, birth_date, gender,  
birth_place, class_id, note, average_score, classification)  
VALUES  
( 'K1101', 'Nguyen Van', 'A', '1986-01-27', 'Nam', 'TpHCM', 'K11', NULL, NULL, NULL),  
( 'K1102', 'Tran Ngoc', 'Han', '1986-03-14', 'Nu', 'Kien Giang', 'K11', NULL, NULL, NULL),  
( 'K1103', 'Ha Duy', 'Lap', '1986-04-18', 'Nam', 'Nghe An', 'K11', NULL, NULL, NULL),  
( 'K1104', 'Tran Ngoc', 'Linh', '1986-03-30', 'Nu', 'Tay Ninh', 'K11', NULL, NULL, NULL),  
( 'K1105', 'Tran Minh', 'Long', '1986-02-27', 'Nam', 'TpHCM', 'K11', NULL, NULL, NULL),  
( 'K1106', 'Le Nhat', 'Minh', '1986-01-24', 'Nam', 'TpHCM', 'K11', NULL, NULL, NULL),  
( 'K1107', 'Nguyen Nhu', 'Nhut', '1986-01-27', 'Nam', 'Ha Noi', 'K11', NULL, NULL, NULL),  
( 'K1108', 'Nguyen Manh', 'Tam', '1986-02-27', 'Nam', 'Kien Giang', 'K11', NULL, NULL, NULL),  
( 'K1109', 'Phan Thi Thanh', 'Tam', '1986-01-27', 'Nu', 'Vinh Long', 'K11', NULL, NULL, NULL),  
( 'K1110', 'Le Hoai', 'Thuong', '1986-05-02', 'Nu', 'Can Tho', 'K11', NULL, NULL, NULL),  
( 'K1111', 'Le Ha', 'Vinh', '1986-12-25', 'Nam', 'Vinh Long', 'K11', NULL, NULL, NULL),  
( 'K1201', 'Nguyen Van', 'B', '1986-11-02', 'Nam', 'TpHCM', 'K12', NULL, NULL, NULL),  
( 'K1202', 'Nguyen Thi Kim', 'Duyen', '1986-01-18', 'Nu', 'TpHCM', 'K12', NULL, NULL, NULL),  
( 'K1203', 'Tran Thi Kim', 'Duyen', '1986-09-17', 'Nu', 'TpHCM', 'K12', NULL, NULL, NULL),  
( 'K1204', 'Truong My', 'Hanh', '1986-05-19', 'Nu', 'Dong Nai', 'K12', NULL, NULL, NULL),  
( 'K1205', 'Nguyen Thanh', 'Nam', '1986-04-17', 'Nam', 'TpHCM', 'K12', NULL, NULL, NULL),  
( 'K1206', 'Nguyen Thi Truc', 'Thanh', '1986-04-03', 'Nu', 'Kien Giang', 'K12', NULL, NULL, NULL),  
( 'K1207', 'Tran Thi Bich', 'Thuy', '1986-08-02', 'Nu', 'Nghe An', 'K12', NULL, NULL, NULL),  
( 'K1208', 'Huynh Thi Kim', 'Trieu', '1986-08-04', 'Nu', 'Tay Ninh', 'K12', NULL, NULL, NULL),  
( 'K1209', 'Pham Thanh', 'Trieu', '1986-02-23', 'Nam', 'TpHCM', 'K12', NULL, NULL, NULL),  
( 'K1210', 'Ngo Thanh', 'Tuan', '1986-02-14', 'Nam', 'TpHCM', 'K12', NULL, NULL, NULL),  
( 'K1211', 'Do Thi', 'Xuan', '1986-09-03', 'Nu', 'Ha Noi', 'K12', NULL, NULL, NULL),  
( 'K1212', 'Le Thi Phi', 'Yen', '1986-12-03', 'Nu', 'TpHCM', 'K12', NULL, NULL, NULL),  
( 'K1301', 'Nguyen Thi Kim', 'Cuc', '1986-09-06', 'Nu', 'Kien Giang', 'K13', NULL, NULL, NULL),  
  
( 'K1302', 'Truong Thi My', 'Hien', '1986-03-18', 'Nu', 'Nghe An', 'K13', NULL, NULL, NULL),  
( 'K1303', 'Le Duc', 'Hien', '1986-03-21', 'Nam', 'Tay Ninh', 'K13', NULL, NULL, NULL),  
( 'K1304', 'Le Quang', 'Hien', '1986-04-18', 'Nam', 'TpHCM', 'K13', NULL, NULL, NULL),  
( 'K1305', 'Le Thi', 'Huong', '1986-03-27', 'Nu', 'TpHCM', 'K13', NULL, NULL, NULL),  
( 'K1306', 'Nguyen Thai', 'Huu', '1986-03-30', 'Nam', 'Ha Noi', 'K13', NULL, NULL, NULL),  
( 'K1307', 'Tran Minh', 'Man', '1986-05-28', 'Nam', 'TpHCM', 'K13', NULL, NULL, NULL),  
( 'K1308', 'Nguyen Hieu', 'Nghia', '1986-08-04', 'Nam', 'Kien Giang', 'K13', NULL, NULL, NULL),  
( 'K1309', 'Nguyen Trung', 'Nghia', '1987-01-18', 'Nam', 'Nghe An', 'K13', NULL, NULL, NULL),  
( 'K1310', 'Tran Thi Hong', 'Tham', '1986-04-22', 'Nu', 'Tay Ninh', 'K13', NULL, NULL, NULL),  
( 'K1311', 'Tran Minh', 'Thuc', '1986-04-04', 'Nam', 'TpHCM', 'K13', NULL, NULL, NULL),  
( 'K1312', 'Nguyen Thi Kim', 'Yen', '1986-07-09', 'Nu', 'TpHCM', 'K13', NULL, NULL, NULL);
```



```
-- insert data into EXAM_RESULTS table
```

```
INSERT INTO exam_results (student_id, subject_id, attempt_number, exam_date, score, result)
VALUES
```

```
('K1101', 'CSDL', 1, '2006-07-20', 10, 'Dat'),
('K1101', 'CTDLGT', 1, '2006-12-28', 9, 'Dat'),
('K1101', 'THDC', 1, '2006-05-20', 9, 'Dat'),
('K1101', 'CTRR', 1, '2006-05-24', 9.5, 'Dat'),
('K1102', 'CSDL', 1, '2006-07-20', 4, 'Khong Dat'),
('K1102', 'CSDL', 2, '2006-07-27', 4.25, 'Khong Dat'),
('K1102', 'CSDL', 3, '2006-08-10', 4.5, 'Khong Dat'),
('K1102', 'CTDLGT', 1, '2006-12-28', 4.5, 'Khong Dat'),
('K1102', 'CTDLGT', 2, '2007-01-05', 4, 'Khong Dat'),
('K1102', 'CTDLGT', 3, '2007-01-15', 6, 'Dat'),
('K1102', 'THDC', 1, '2006-05-20', 5, 'Dat'),
('K1102', 'CTRR', 1, '2006-05-24', 7, 'Dat'),
('K1103', 'CSDL', 1, '2006-07-20', 3.5, 'Khong Dat'),
('K1103', 'CSDL', 2, '2006-07-27', 8.25, 'Dat'),
('K1103', 'CTDLGT', 1, '2006-12-28', 7, 'Dat'),
('K1103', 'THDC', 1, '2006-05-20', 8, 'Dat'),
('K1103', 'CTRR', 1, '2006-05-24', 6.5, 'Dat'),
('K1104', 'CSDL', 1, '2006-07-20', 3.75, 'Khong Dat'),
('K1104', 'CTDLGT', 1, '2006-12-28', 4, 'Khong Dat'),
('K1104', 'THDC', 1, '2006-05-20', 4, 'Khong Dat'),
('K1104', 'CTRR', 1, '2006-05-24', 4, 'Khong Dat'),
('K1104', 'CTRR', 2, '2006-06-01', 3.5, 'Khong Dat'),
('K1104', 'CTRR', 3, '2006-06-30', 4, 'Khong Dat'),
('K1201', 'CSDL', 1, '2006-07-20', 6, 'Dat'),
('K1201', 'CTDLGT', 1, '2006-12-28', 5, 'Dat'),
('K1201', 'THDC', 1, '2006-05-20', 8.5, 'Dat'),
('K1201', 'CTRR', 1, '2006-05-24', 9, 'Dat'),
('K1202', 'CSDL', 1, '2006-07-20', 8, 'Dat'),
('K1202', 'CTDLGT', 1, '2006-12-28', 4, 'Khong Dat'),
('K1202', 'CTDLGT', 2, '2007-05-01', 5, 'Dat'),
('K1202', 'THDC', 1, '2006-05-20', 4, 'Khong Dat'),
('K1202', 'THDC', 2, '2006-05-27', 4, 'Khong Dat'),
('K1202', 'CTRR', 1, '2006-05-24', 3, 'Khong Dat'),
('K1202', 'CTRR', 2, '2006-06-01', 4, 'Khong Dat'),
('K1202', 'CTRR', 3, '2006-06-30', 6.25, 'Dat'),
('K1203', 'CSDL', 1, '2006-07-20', 9.25, 'Dat'),
('K1203', 'CTDLGT', 1, '2006-12-28', 9.5, 'Dat'),
('K1203', 'THDC', 1, '2006-05-20', 10, 'Dat'),
('K1203', 'CTRR', 1, '2006-05-24', 10, 'Dat'),
('K1204', 'CSDL', 1, '2006-07-20', 8.5, 'Dat'),
('K1204', 'CTDLGT', 1, '2006-12-28', 6.75, 'Dat'),
('K1204', 'THDC', 1, '2006-05-20', 4, 'Khong Dat'),
('K1204', 'CTRR', 1, '2006-05-24', 6, 'Dat'),
('K1301', 'CSDL', 1, '2006-12-20', 4.25, 'Khong Dat'),
('K1301', 'CTDLGT', 1, '2006-07-25', 8, 'Dat'),
```

```

('K1301', 'THDC', 1, '2006-05-20', 7.75, 'Dat'),
('K1301', 'CTRR', 1, '2006-05-24', 8, 'Dat'),
('K1302', 'CSDL', 1, '2006-12-20', 6.75, 'Dat'),
('K1302', 'CTDLGT', 1, '2006-07-25', 5, 'Dat'),
('K1302', 'THDC', 1, '2006-05-20', 8, 'Dat'),
('K1302', 'CTRR', 1, '2006-05-24', 8.5, 'Dat'),
('K1303', 'CSDL', 1, '2006-12-20', 4, 'Khong Dat'),
('K1303', 'CTDLGT', 1, '2006-07-25', 4.5, 'Khong Dat'),
('K1303', 'CTDLGT', 2, '2006-08-07', 4, 'Khong Dat'),
('K1303', 'CTDLGT', 3, '2006-08-15', 4.25, 'Khong Dat'),
('K1303', 'THDC', 1, '2006-05-20', 4.5, 'Khong Dat'),
('K1303', 'CTRR', 1, '2006-05-24', 3.25, 'Khong Dat'),
('K1303', 'CTRR', 2, '2006-06-01', 5, 'Dat'),
('K1304', 'CSDL', 1, '2006-12-20', 7.75, 'Dat'),
('K1304', 'CTDLGT', 1, '2006-07-25', 9.75, 'Dat'),
('K1304', 'THDC', 1, '2006-05-20', 5.5, 'Dat'),
('K1304', 'CTRR', 1, '2006-05-24', 5, 'Dat'),
('K1305', 'CSDL', 1, '2006-12-20', 9.25, 'Dat'),
('K1305', 'CTDLGT', 1, '2006-07-25', 10, 'Dat'),
('K1305', 'THDC', 1, '2006-05-20', 8, 'Dat'),
('K1305', 'CTRR', 1, '2006-05-24', 10, 'Dat');

-- Re-enable foreign key checks
SET FOREIGN_KEY_CHECKS = 1;
SET SQL_SAFE_UPDATES = 1;

```

### ***VIII. Truy vấn dữ liệu trong cơ sở giáo vụ***

Để khai thác dữ liệu lưu trữ trong hệ thống, chúng ta có thể sử dụng một số câu lệnh truy vấn SQL để truy cập thông tin mong muốn.

Ví dụ một số lệnh truy vấn :

1. In ra danh sách (mã học viên, họ tên, ngày sinh, mã lớp) lớp trưởng của các lớp.

```

SELECT
    s.student_id AS "Student ID",
    CONCAT(s.last_name, ' ', s.first_name) AS "Full Name",
    s.birth_date AS "Birth Date",
    c.class_id AS "Class ID"
FROM
    student s
JOIN
    class c ON s.student_id = c.class_leader_id;

```

Kết quả :

Student ID	Full Name	Birth Date	Class ID
K1108	Nguyen Manh Tam	1986-02-27	K11
K1205	Nguyen Thanh Nam	1986-04-17	K12
K1305	Le Thi Huong	1986-03-27	K13

2. In ra danh sách những học viên (mã học viên, họ tên) và những môn học mà học viên đó thi lần thứ nhất đã đạt

```
SELECT
    e.student_id AS "Student ID",
    CONCAT(s.last_name, ' ', s.first_name) AS "Full Name",
    e.subject_id AS "Subject ID"
FROM
    exam_results e
JOIN
    student s ON e.student_id = s.student_id
WHERE
    e.attempt_number = 1 AND e.score >= 5;
```

Kết quả :

Student ID	Full Name	Subject ID
K1101	Nguyen Van A	CSDL
K1101	Nguyen Van A	CTDLGT
K1101	Nguyen Van A	CTRR
K1101	Nguyen Van A	THDC
K1102	Tran Ngoc Han	CTRR
K1102	Tran Ngoc Han	THDC
K1103	Ha Duong Lan	CTDLGT

3. In ra danh sách những môn học (mã môn học, tên môn học) phải học liền trước môn “Co So Du Lieu”.

```
SELECT
    p.prerequisite_subject_id AS "Subject ID",
    su.subject_name AS "Subject Name"
FROM
    prerequisite p
JOIN
    subject su ON p.prerequisite_subject_id = su.subject_id
WHERE
    p.subject_id = 'CSDL';
```

Kết quả :

	Subject ID	Subject Name
▶	CTRR	Cau truc roi rac

4. Tìm những học viên (mã học viên, họ tên) thi không đạt môn CSDL ở lần thi thứ 1 nhưng chưa thi lại môn này.

```
SELECT
    s.student_id AS "Student ID",
    CONCAT(s.last_name, ' ', s.first_name) AS "Full Name"
FROM
    exam_results e
JOIN
    student s ON e.student_id = s.student_id
WHERE
    e.subject_id = 'CSDL'
    AND e.attempt_number = 1
    AND e.score < 5
    AND NOT EXISTS (
        SELECT 1
        FROM exam_results e2
        WHERE e2.student_id = e.student_id
        AND e2.subject_id = 'CSDL'
        AND e2.attempt_number > 1
    );
```

Kết quả :

	Student ID	Full Name
▶	K1104	Tran Ngoc Linh
	K1301	Nguyen Thi Kim Cuc
	K1303	Le Duc Hien

5. Tìm giáo viên (mã giáo viên, họ tên) không được phân công giảng dạy bất kỳ môn học nào.

```
SELECT te.teacher_id AS "Teacher ID",
    te.full_name AS "Full Name"
FROM teacher te
WHERE NOT EXISTS (
    SELECT 1
    FROM teaching t
    WHERE t.teacher_id = te.teacher_id
);
```

Kết quả :

	Teacher ID	Full Name
▶	GV01	Ho Thanh Son
	GV03	Do Nghiem Phung
	GV10	Le Tran Anh Loan
	GV11	Ho Thanh Tung
	GV12	Tran Van Anh
	GV13	Nguyen Linh Dan
	GV14	Truong Minh Chau

## ***IX. Trigger, Transaction và Procedure***

### ***1. Trigger***

\*Ta có thể viết trigger để đảm bảo trong quá trình chèn hoặc cập nhật dữ liệu, lớp trưởng của một lớp phải là sinh viên của lớp đó :

```
CREATE TRIGGER trg_insert_class_leader
BEFORE INSERT ON class
FOR EACH ROW
BEGIN
    IF (SELECT COUNT(*)
        FROM student
        WHERE student.student_id = NEW.class_leader_id
            AND student.class_id != NEW.class_id) > 0 THEN
        SIGNAL SQLSTATE '45000'
            SET MESSAGE_TEXT = 'Lớp trưởng của một lớp phải là học viên của lớp đó';
    END IF;
END$$
```

```
CREATE TRIGGER trg_update_class_leader
BEFORE UPDATE ON class
FOR EACH ROW
BEGIN
    IF (SELECT COUNT(*)
        FROM student
        WHERE student.student_id = NEW.class_leader_id
            AND student.class_id != NEW.class_id) > 0 THEN
        SIGNAL SQLSTATE '45000'
            SET MESSAGE_TEXT = 'Lớp trưởng của một lớp phải là học viên của lớp đó';
    END IF;
END$$
```

### ***2. Transaction (có sử dụng ROLLBACK)***

- Thêm môn mới CSTTNT và chèn các dữ liệu liên quan :

```

START TRANSACTION;

INSERT INTO subject (subject_id, subject_name, theory_credit, practical_credit, department_id)
VALUES ('CSTTNT', 'Co so tri tue nhan tao', 3, 0, 'KHMT');

INSERT INTO prerequisite (subject_id, prerequisite_subject_id)
VALUES ('CSTTNT', 'CTDLGT');

INSERT INTO teaching (class_id, subject_id, teacher_id, semester, year, start_date, end_date)
VALUES ('K12', 'CSTTNT', 'GV04', 1, 2007, '2006-01-08', '2006-12-15');

ROLLBACK;

```

- Trong trường hợp giao dịch thất bại, lệnh ROLLBACK sẽ được thực thi để hủy bỏ các thao tác.

### 3. Procedure

\*Viết procedure để thêm dữ liệu một sinh viên mới

```

DELIMITER $$
CREATE PROCEDURE ThemSinhVien(
in MSV char(5),
in Last_Name varchar(40),
in First_Name varchar(10),
in birthday date,
in sex varchar(3),
in birthplace varchar(40),
in Class_ID char(3),
in Note varchar(100),
in GPA decimal(4,2),
in Classification varchar(10)
)
BEGIN
IF NOT EXISTS (SELECT 1 FROM class WHERE class_id = Class_ID) THEN
    SIGNAL SQLSTATE '45000'
    SET MESSAGE_TEXT = 'Lớp không tồn tại!';
END IF;

IF GPA < 0 OR GPA > 4 THEN
    SIGNAL SQLSTATE '45000'
    SET MESSAGE_TEXT = 'GPA không hợp lệ! Giá trị phải nằm trong khoảng 0-4.';
END IF;

INSERT INTO student(student_id, last_name, first_name, birth_date, gender,
birth_place, class_id, note, average_score, classification)
VALUES(MSV, Last_Name, First_Name, birthday, sex, birthplace, Class_ID, Note, GPA, Classification);
END $$
DELIMITER ;

```

\* Viết Procedure cập nhật trạng thái của các sinh viên thi lần 3 dưới 5 điểm :



```

CREATE PROCEDURE sp_update_ban_examNote()
BEGIN
    UPDATE student
    SET note = 'Cam thi'
    WHERE student_id IN (
        SELECT student_id
        FROM exam_results
        WHERE attempt_number = 3 AND score < 5
    );
END //

```

## ***X. Tổng kết***

- Hệ thống đã tái hiện trực quan vận hành dữ liệu của một cơ sở giáo dục, có thể đảm bảo các nghiệp vụ cơ bản trong việc quản lý dữ liệu sinh viên, giảng viên, các lớp, các môn học, dữ liệu giảng dạy, điểm thi và tình trạng của sinh viên, giảng viên.
- Tính tổ chức và liên kết chặt chẽ cho phép việc chỉnh sửa và khai thác dữ liệu từ hệ thống trở nên dễ dàng, tiện lợi. Giáo viên dễ dàng nắm bắt được thông tin giảng dạy, sinh viên đăng ký; sinh viên có thể chủ động nắm bắt được thông tin môn học, các lớp học phân để đăng ký theo mong muốn; nhà trường quản lý được tình hình học tập của sinh viên để đưa ra cảnh báo kịp thời.