**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.

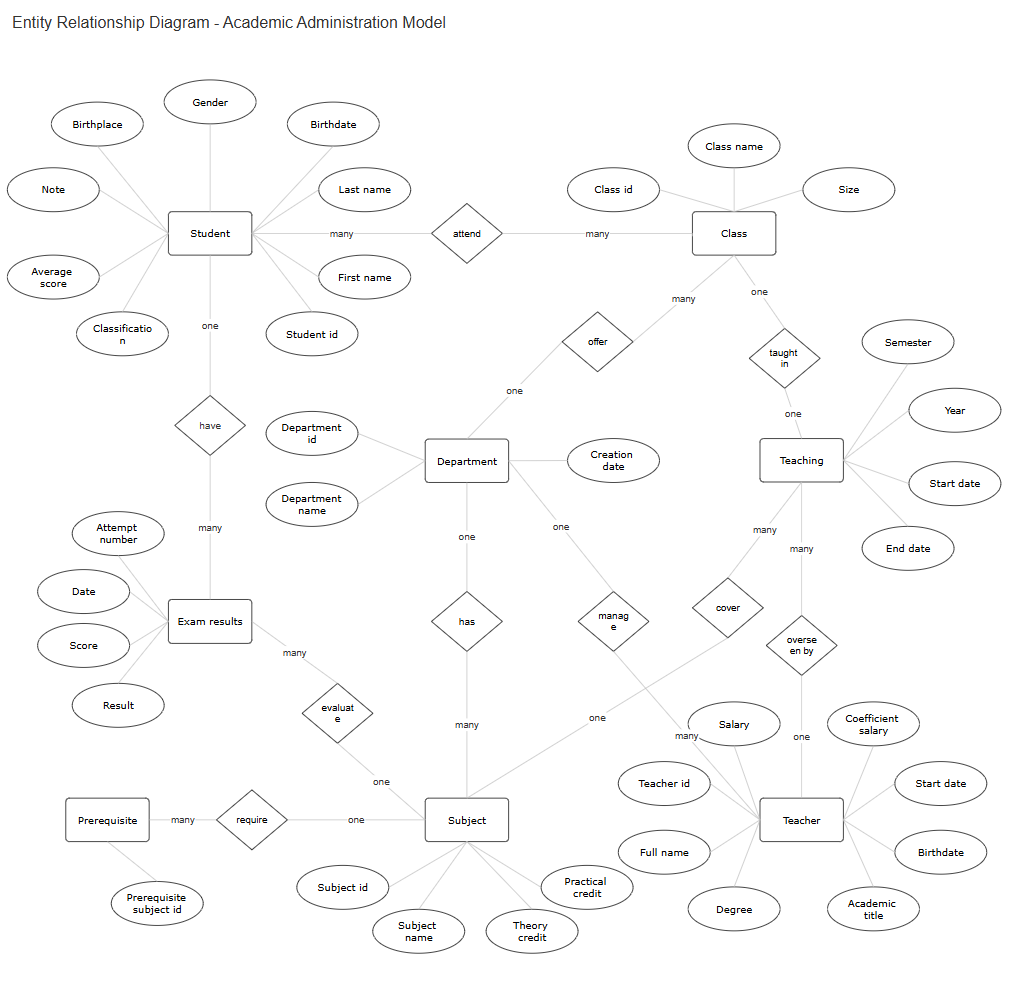
***II. Nghiệp vụ của hệ thống***

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 :



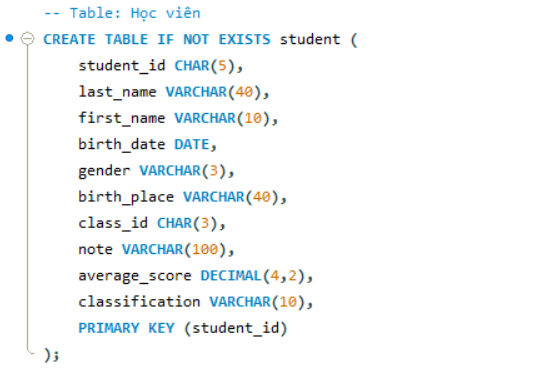
***IV. Mô hình quan hệ chuyển từ mô hình ER***

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

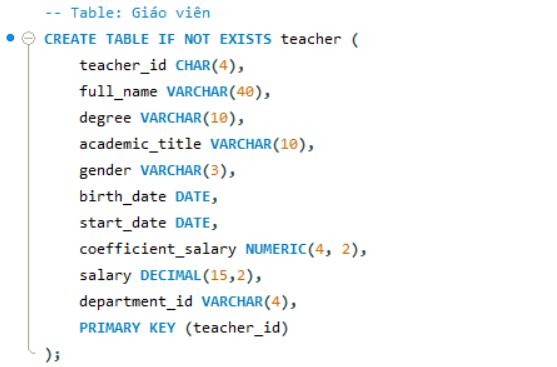
+ Tạo một database mới mang tên academic\_administration :



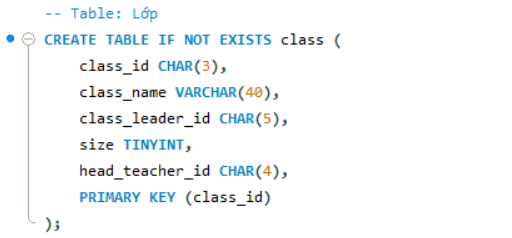
+ Tạo bảng lưu các thuộc tính cho thực thể sinh viên :



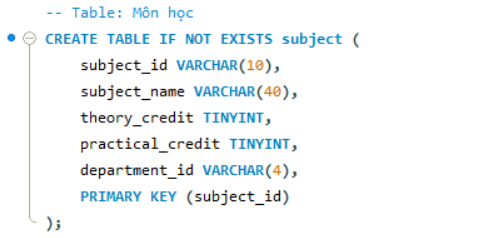
+ Tạo bảng lưu các thuộc tính cho giáo viên :



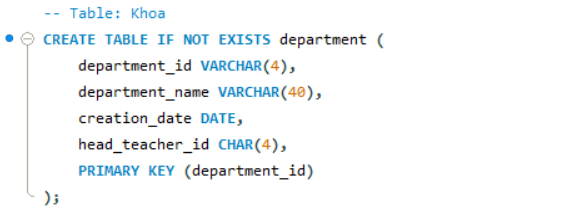
+ Tạo bảng lưu thực thể lớp học :



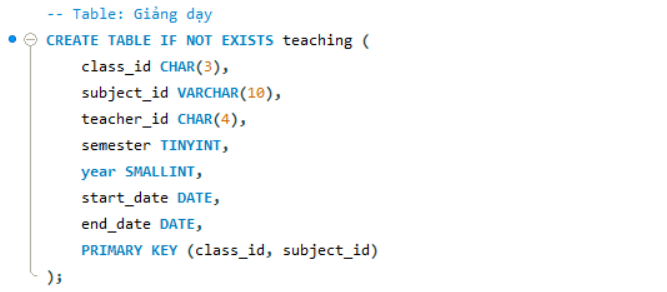
+ Tạo bảng lưu thực thể môn học :



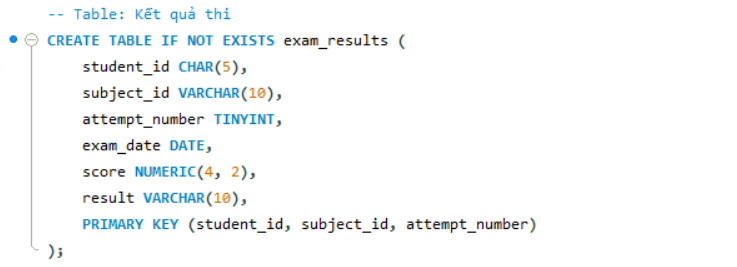
+ Tạo bảng cho thực thể khoa :



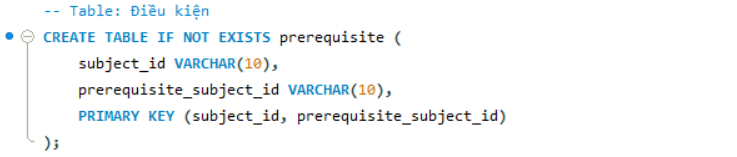
+ Tạo bảng lưu dữ liệu giảng dạy :



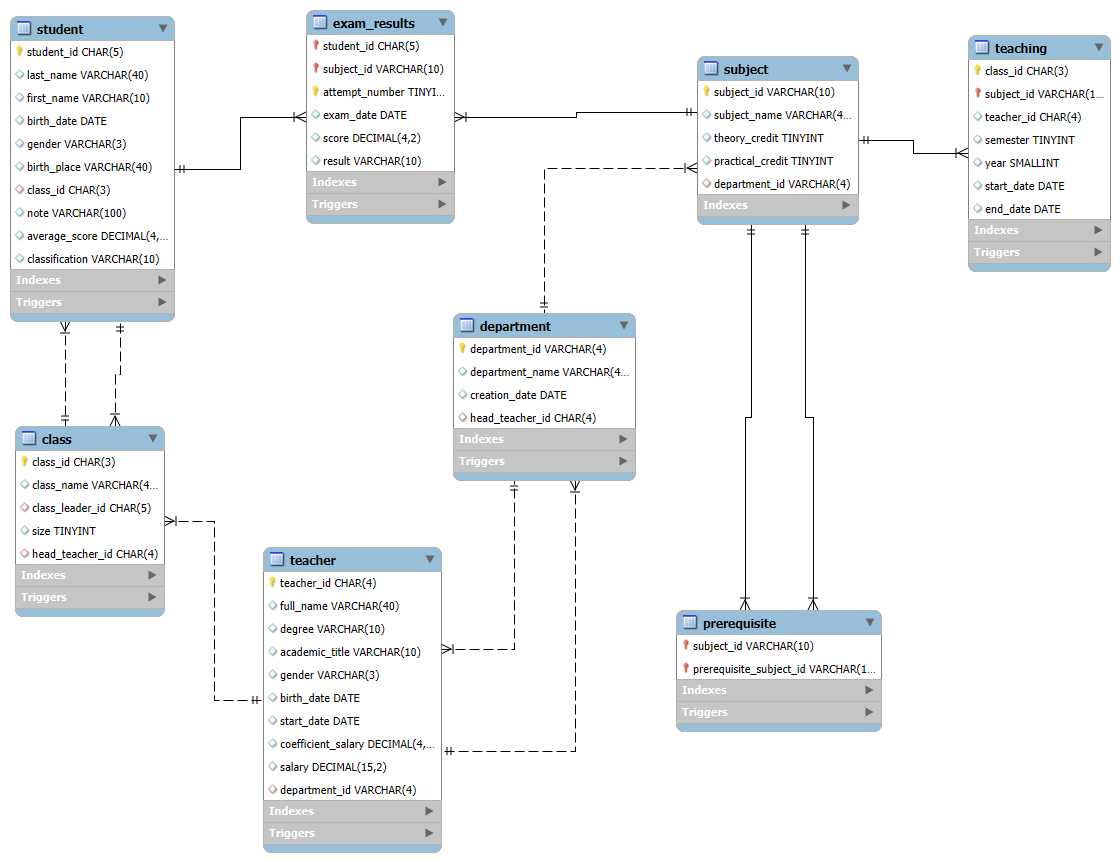
+ Tạo bảng lưu thông tin các kết quả thi :



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



• 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)

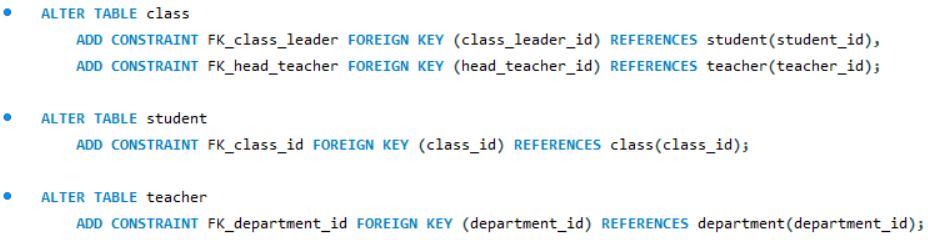
***V. Từ điển dữ liệu***

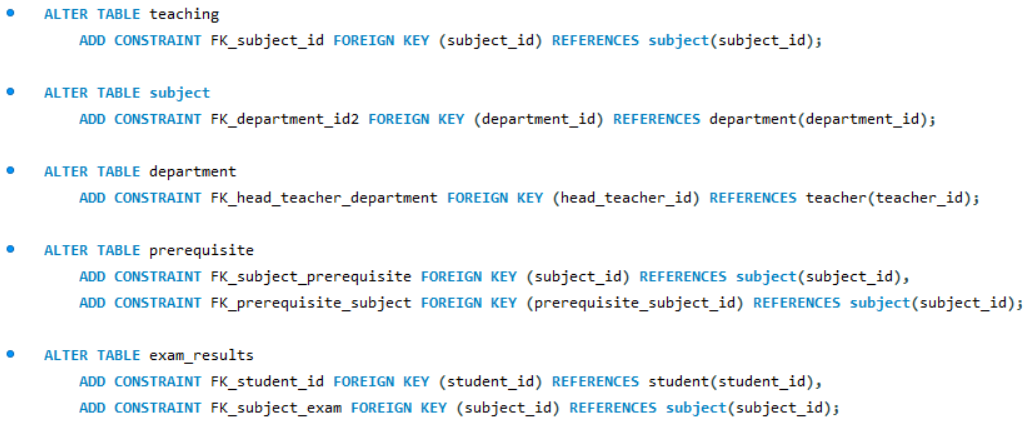
|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Mô tả | Kiểu dữ liệu | Dung lượng tối đa |
| student\_id | Mã sinh viên | CHAR | 5 |
| last\_name | Tên họ của cá nhân | VARCHAR | 40 |
| first\_name | Tên đầu của cá nhân | VARCHAR | 10 |
| birth\_date | Ngày sinh của cá nhân | DATE |  |
| gender | Giới tính cá nhân | VARCHAR | 3 |
| birth\_place | Nơi sinh | VARCHAR | 40 |
| class\_id | Mã lớp học | CHAR | 3 |
| note | Ghi chú về cá nhân | VARCHAR | 100 |
| average\_score | Điểm trung bình | DECIMAL | (4,2) |
| classification | Dữ liệu phân loại sinh viên | VARCHAR | 10 |
| subject\_id | Mã môn học | VARCHAR | 10 |
| teacher\_id | Mã giảng viên | CHAR | 4 |
| semester | Học kỳ | TINYINT |  |
| year | Năm học | SMALLINT |  |
| start\_date | Ngày bắt đầu giảng dạy | DATE |  |
| end\_date | Ngày kết thúc giảng dạy | DATE |  |
| department\_id | Mã khoa | VARCHAR | 4 |
| department\_name | Tên khoa | VARCHAR | 40 |
| creation\_date | Ngày thành lập khoa | DATE |  |
| head\_teacher\_id | Mã giảng viên phụ trách lớp | CHAR | 4 |
| subject\_name | Tên môn học | VARCHAR | 40 |
| theory\_credit | Số tín chỉ lý thuyết | TINYINT |  |
| practical\_credit | Số tín chỉ thực hành | TINYINT |  |
| prerequisite\_subject\_id | Mã môn học tiền đề | VARCHAR | 10 |
| degree | Học vị | VARCHAR | 10 |
| academic\_title | Chức danh khoa học | VARCHAR | 10 |
| coefficient\_salary | Hệ số lương | NUMERIC | (4, 2) |
| salary | Lương | DECIMAL | (15,2) |
| class\_name | Tên lớp học | VARCHAR | 40 |
| class\_leader\_id | Mã sinh viên của lớp trưởng | CHAR | 5 |
| size | Sĩ số lớp | TINYINT |  |
| attempt\_number | Số lần thi | TINYINT |  |
| exam\_date | Ngày thi | DATE |  |
| score | Điểm thi | NUMERIC | (4, 2) |
| result | Kết quả thi | VARCHAR | 10 |

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

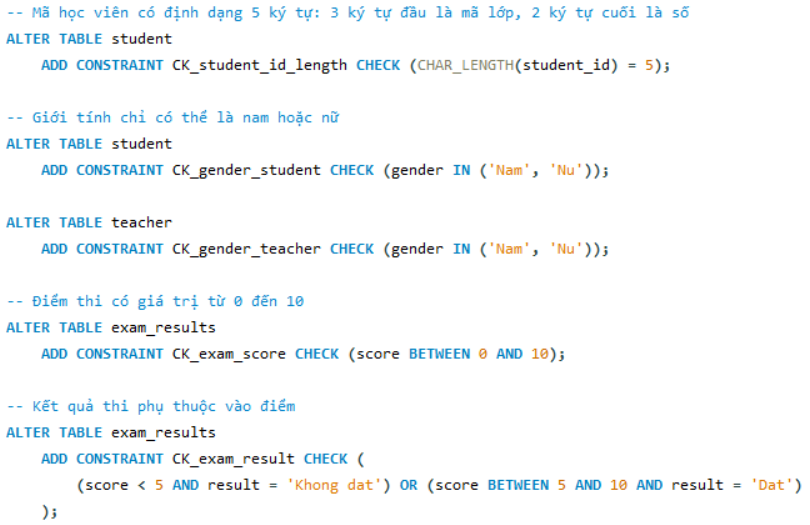
1. Thêm các khóa ngoại và ràng buộc đối với một số thuộc tính

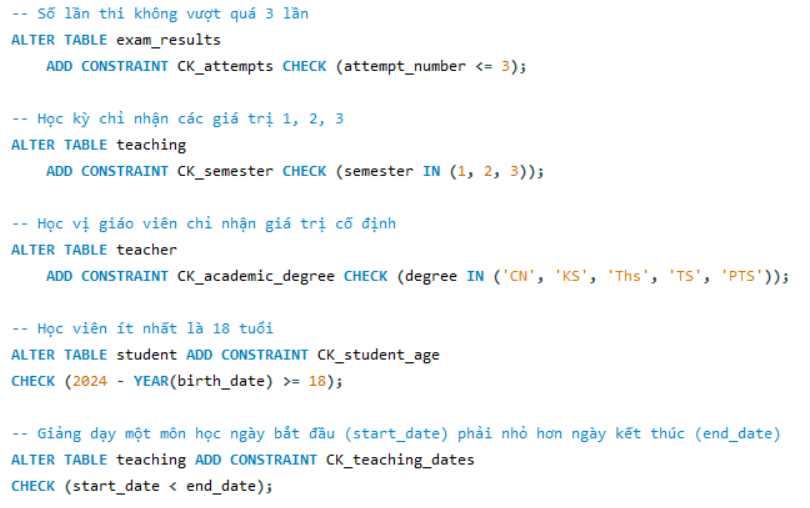
\* Thêm khóa ngoại

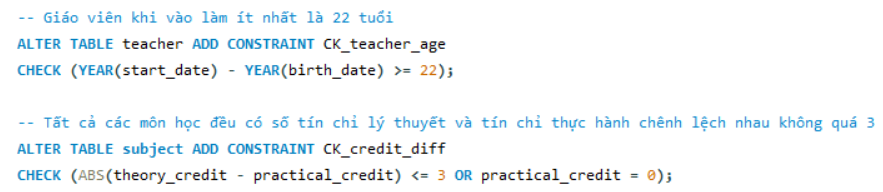




\*Thêm các ràng buộc







**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

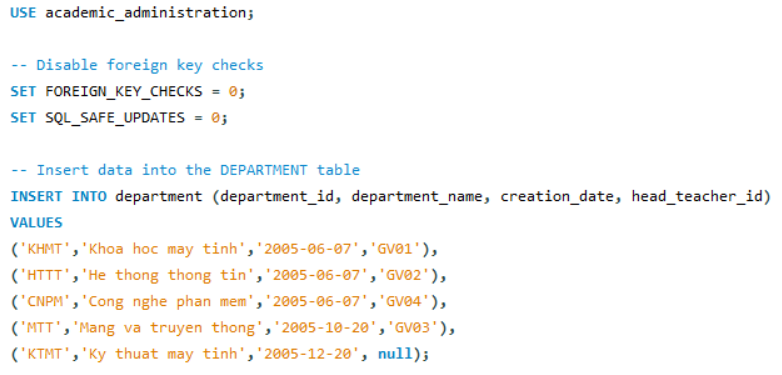
+ Ngày bắt đầu giảng dạy phải nhỏ hơn ngày kết thúc giảng dạy

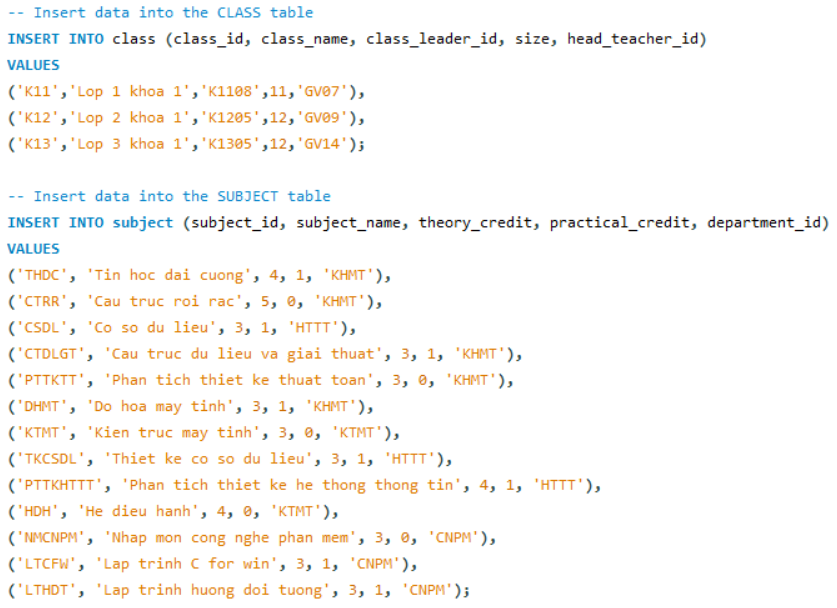
+ Độ tuổi tối thiểu của giảng viên là 22

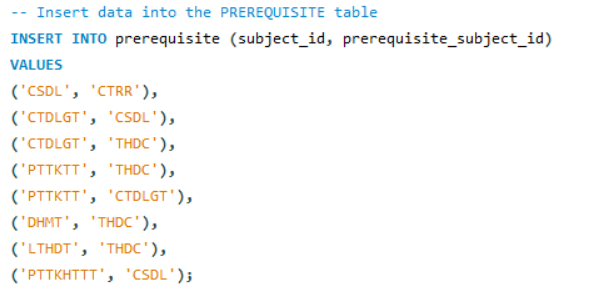
+ Tất cả các môn học đều có số tín chỉ lý thuyết và tín chỉ thực hành chênh lệch nhau không quá 3.

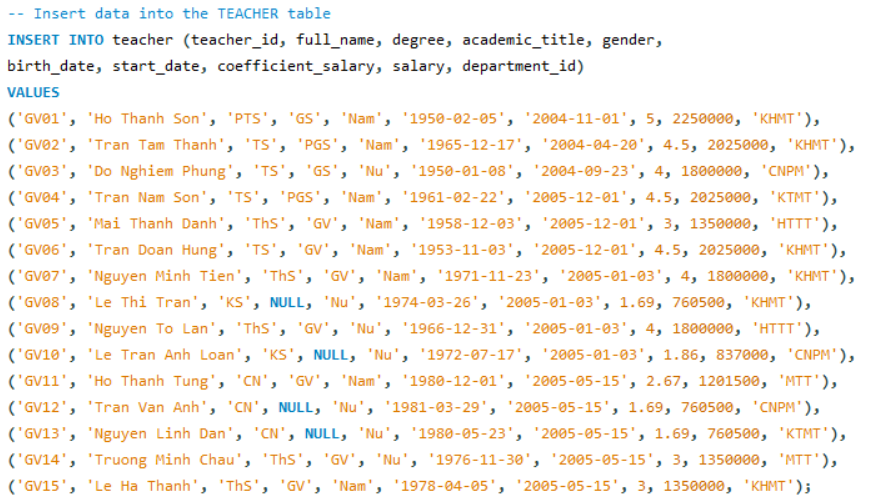
***VII. Thêm dữ liệu phù hợp cho cơ sở dữ liệu quản lý giáo vụ***

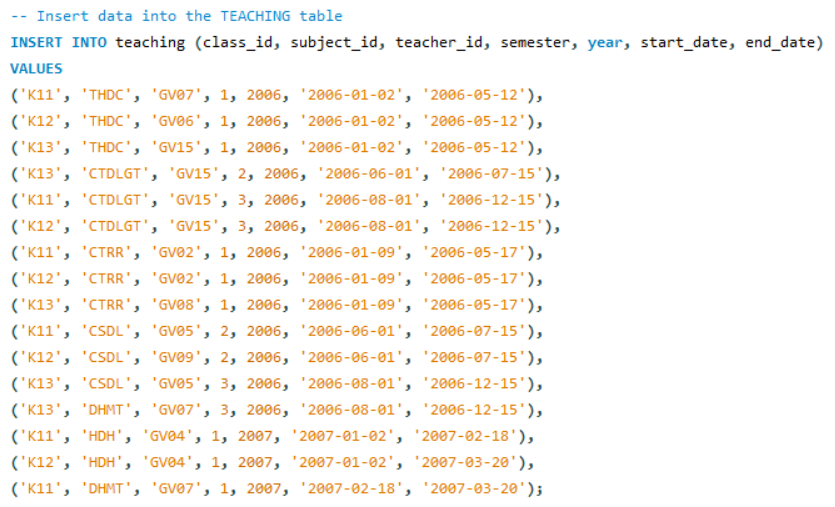
\*Sử dụng các câu lệnh INSERT INTO ….VALUES (….) :

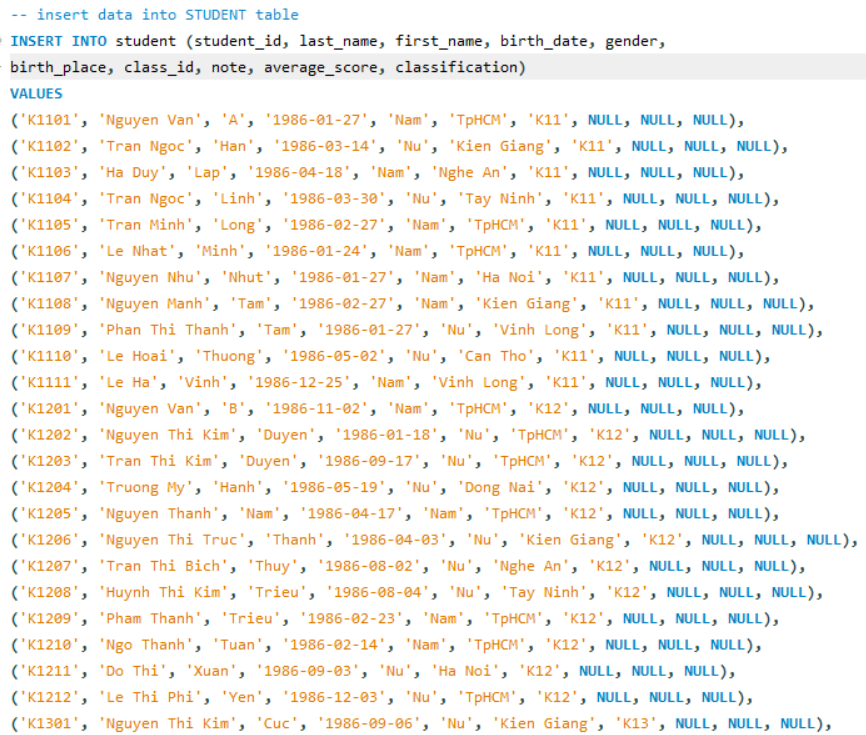




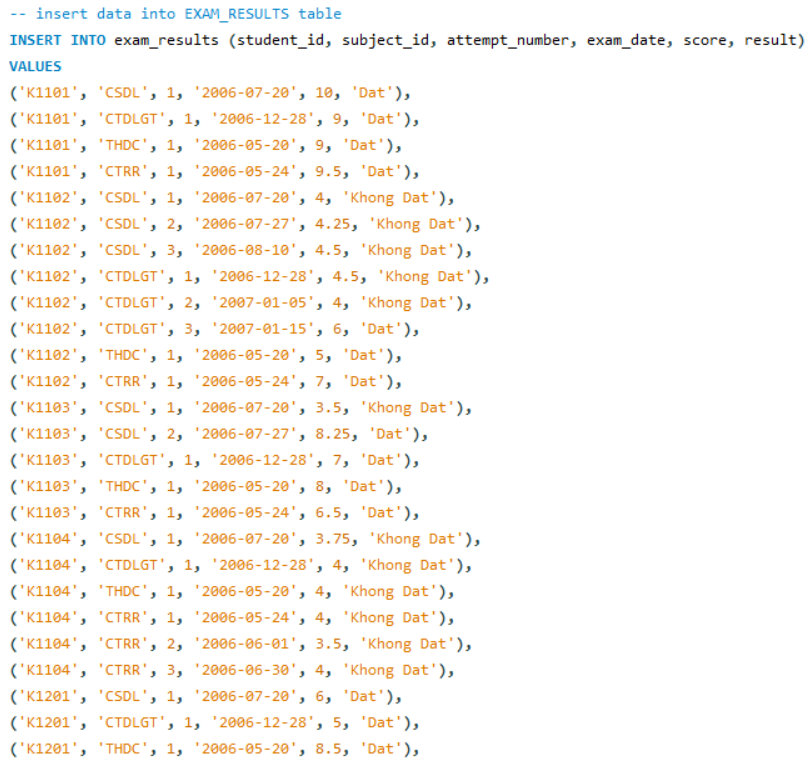


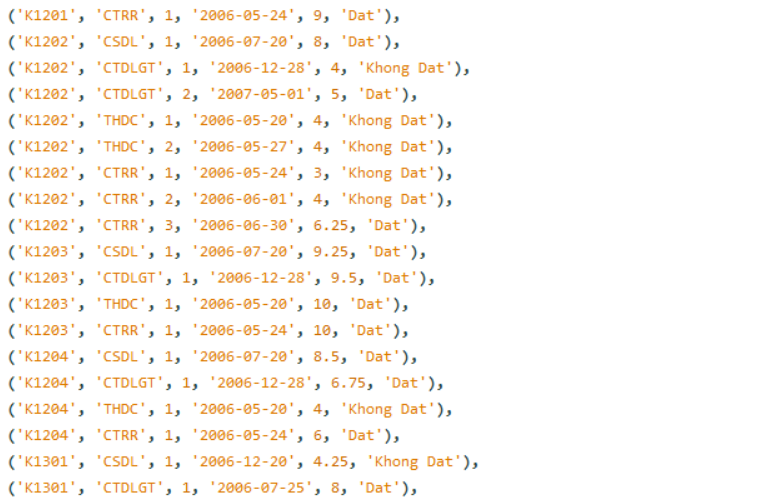


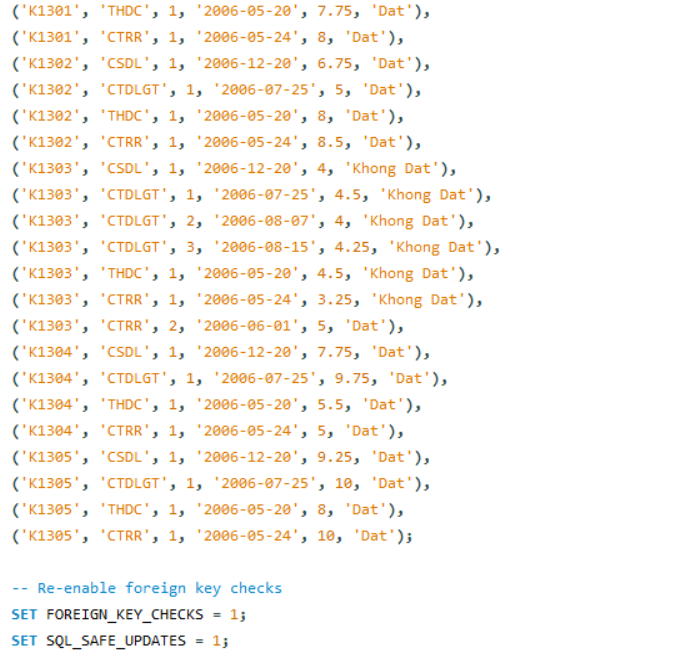










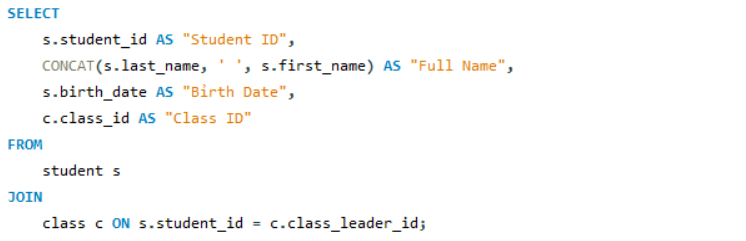


***VIII. Truy vấn dữ liệu trong cơ sở giáo vụ***

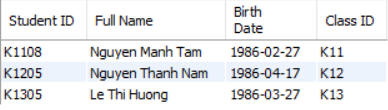
Để khai thác dữ liệu lưu trữ trong hệ thống, chúng ta có thể sử dụng một số câu lệnh truy vấn SQL để truy cập thông tin mong muốn.

Ví dụ một số lệnh truy vấn :

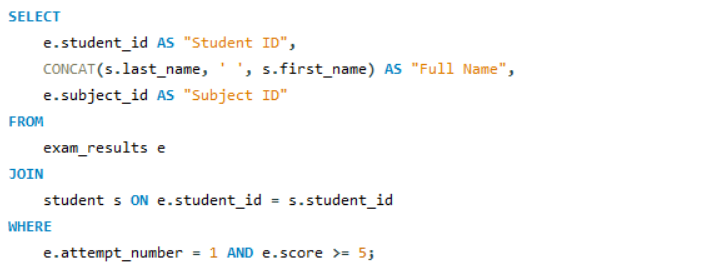
1. In ra danh sách (mã học viên, họ tên, ngày sinh, mã lớp) lớp trưởng của các lớp.



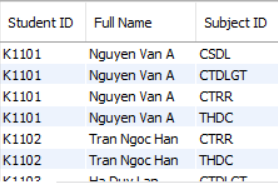
Kết quả :



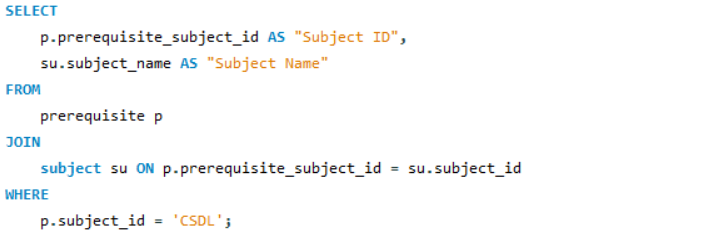
2. In ra danh sách những học viên (mã học viên, họ tên) và những môn học mà học viên đó thi lần thứ nhất đã đạt



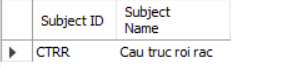
Kết quả :



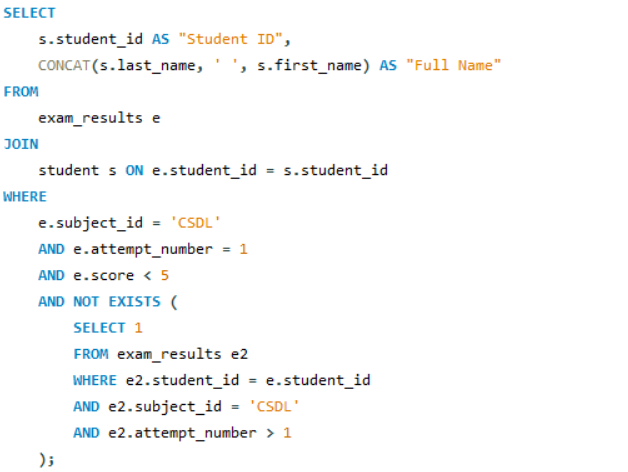
3. In ra danh sách những môn học (mã môn học, tên môn học) phải học liền trước môn “Co So Du Lieu”.



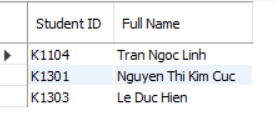
Kết quả :



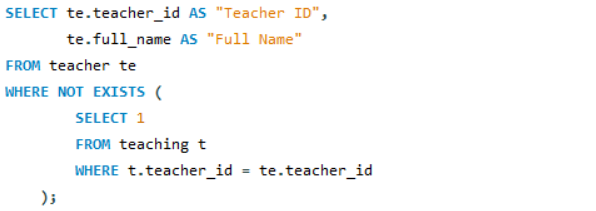
4. Tìm những học viên (mã học viên, họ tên) thi không đạt môn CSDL ở lần thi thứ 1 nhưng chưa thi lại môn này.



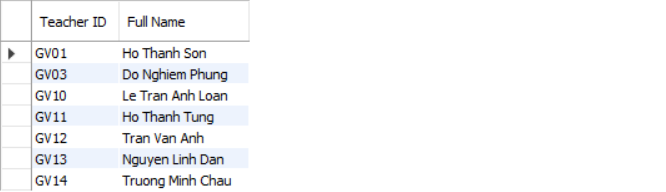
Kết quả :



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



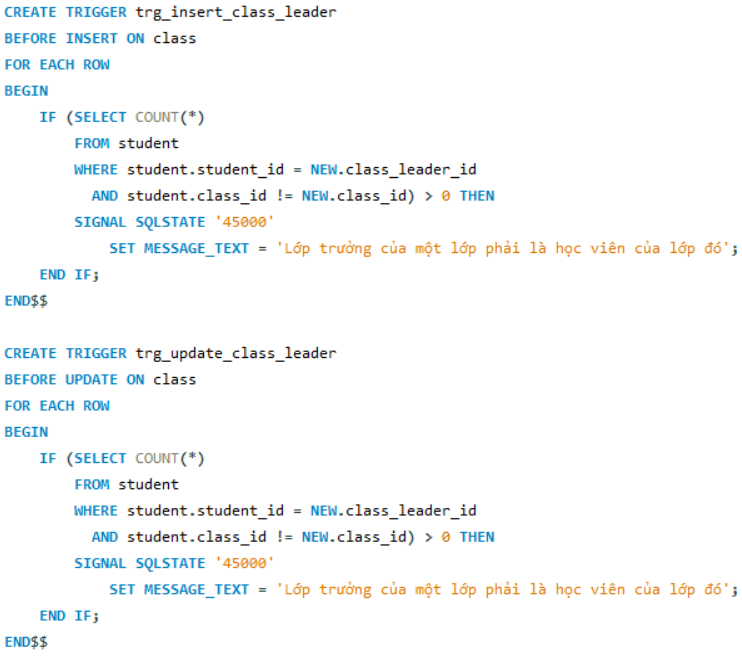
Kết quả :



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

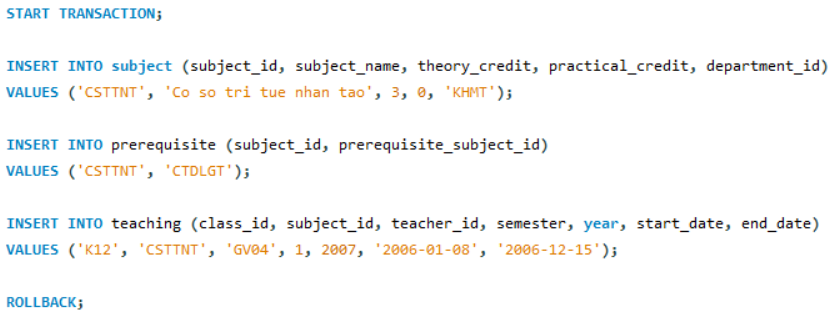
*1. Trigger*

\*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 :



*2. Transaction (có sử dụng ROLLBACK)*

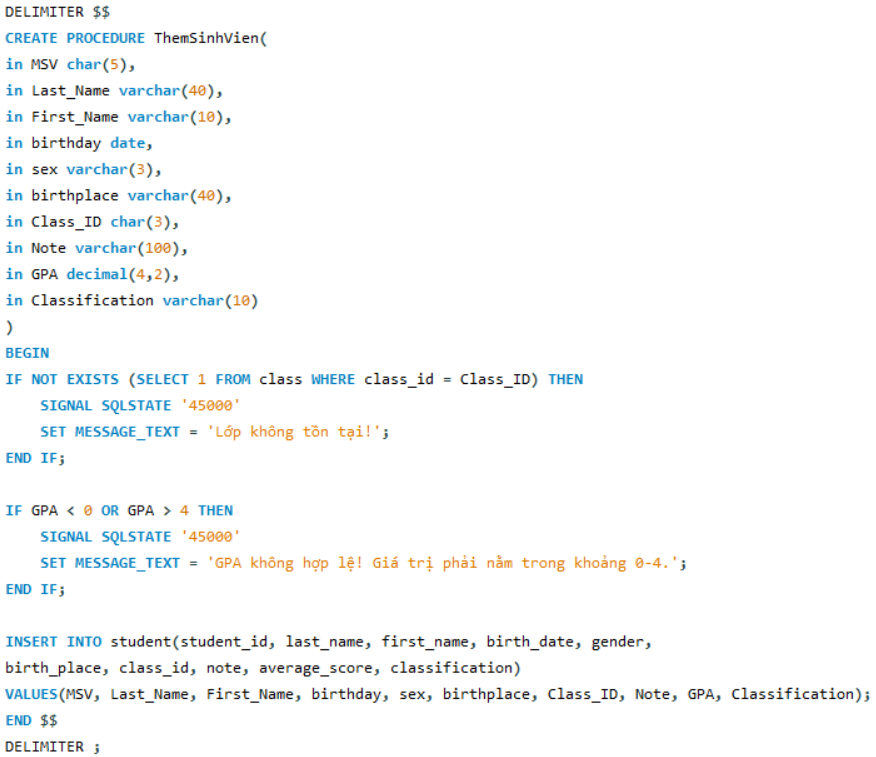
- Thêm môn mới CSTTNT và chèn các dữ liệu liên quan :



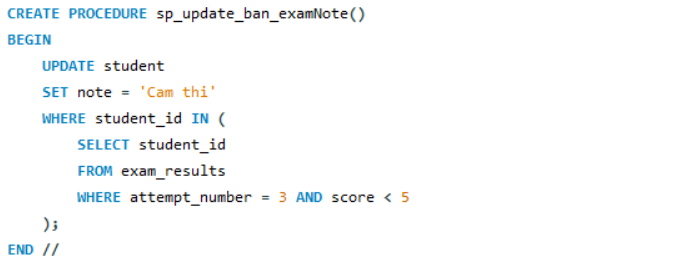
- Trong trường hợp giao dịch thất bại, lệnh ROLLBACK sẽ được thực thi để hủy bỏ các thao tác.

*3. Procedure*

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



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



***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 phần 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.