TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP

KHOA ĐIỆN TỬ

BÀI TẬP LỚN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

NGÀNH : KỸ THUẬT MÁY TÍNH

HỆ : ĐẠI HỌC CHÍNH QUY

THÁI NGUYÊN - 2025

TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP
KHOA ĐIỆN TỬ

BÀI TẬP LỚN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

BỘ MÔN : CÔNG NGHỆ THÔNG TIN

GIÁO VIÊN HƯỚNG DẪN	:	ThS.ĐỖ DUY CỐP
HỌ VÀ TÊN SINH VIÊN	:	NGUYỄN THỊ XUÂN PHƯƠNG
LỚP	:	K58KTPM
MSSV	:	K225480106054

THÁI NGUYÊN - 202

TRƯỜNG ĐHKTCN

KHOA ĐIỆN TỬ

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

PHIẾU GHI ĐIỂM

TIỂU LUẬN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

Sinh viên:

Lớp: K58KTP

GVHD: ThS.Đỗ Duy Cốp

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

……

……

……

……

Xếp loại: …… Điểm : ……

Thái Nguyên, ngày….tháng…..năm 20....

GIÁO VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

MỤC LỤC

LỜI NÓI ĐẦU 3
CHƯƠNG 1. BÀI TẬP VỀ NHÀ SỐ 1 4
1. Đề bài 4
2. Bài Làm 4
3. Mã QR Github của bài tập 4
CHƯƠNG 2. BÀI TẬP VỀ NHÀ SỐ 2 15
1. Đề Bài 15
2. Bài Làm 16
2.2 Code 18
3. Mã QR của bài 22
CHƯƠNG 3. BÀI TẬP VỀ NHÀ 3 23
1. Đề bài 23
2. Bài Làm 24
3. Mã QR của bài 29
CHƯƠNG 4. BÀI TẬP VỀ NHÀ 4 30
1. Đề bài 30
2. Bài làm 30
2.1 Code 33
3. Mã QR của bài 35
CHƯƠNG 5. BÀI TẬP VỀ NHÀ 5 36
1. Đề bài 36
2. Bài làm 36
3. Mã QR của bài 44
CHƯƠNG 6: BÀI TẬP VỀ NHÀ 6 45
1. Đề bài 45
2. Bài làm 45
3. Mã QR của bài 52
CHƯƠNG 7: BÀI KIỂM TRA 53
1. Đề bài 53
2. Bài làm 54
3. Mã QR của bài 56
KẾT LUẬN 57

LỜI NÓI ĐẦU

Trong bối cảnh cuộc cách mạng công nghiệp 4.0 đang diễn ra mạnh mẽ, công nghệ thông tin đóng vai trò quan trọng trong hầu hết các lĩnh vực đời sống và sản xuất. Một trong những nền tảng cốt lõi của công nghệ thông tin chính là cơ sở dữ liệu – nơi lưu trữ và xử lý thông tin phục vụ các hệ thống phần mềm. Trong khuôn khổ học phần Hệ quản trị cơ sở dữ liệu, em đã có cơ hội tiếp cận, thực hành và triển khai các hệ thống dữ liệu thực tế, qua đó hiểu rõ hơn về quy trình thiết kế, xây dựng, quản lý và khai thác hiệu quả một cơ sở dữ liệu.

Tiểu luận này tổng hợp lại quá trình học tập và thực hành của em thông qua các bài tập lớn nhỏ với những yêu cầu cụ thể từ giảng viên. Em đã áp dụng các công cụ như SQL Server, Management Studio, kết hợp với kiến thức nền về mô hình dữ liệu, khóa chính - khóa ngoại, trigger, thủ tục (stored procedure),... để hoàn thành nhiệm vụ được giao. Đây cũng là cơ hội quý báu để em rèn luyện kỹ năng lập trình truy vấn, phân tích dữ liệu và xây dựng hệ thống thông tin hiệu quả.

CHƯƠNG 1. BÀI TẬP VỀ NHÀ SỐ 1

1. *Đề bài*

❖ Cài Đặt SQL Sever

2. *Bài Làm*

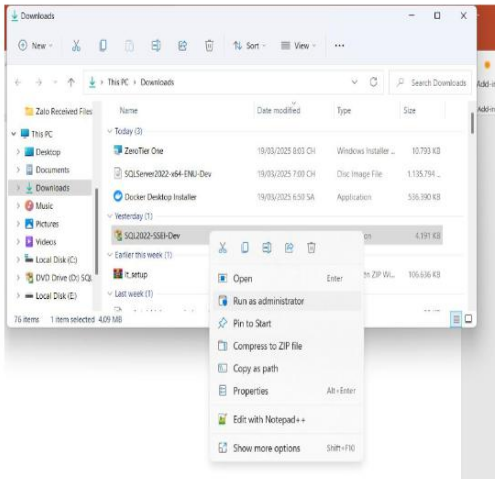
Họ và tên: Nguyễn Thị Xuân Phương

MSSV: K225480106054. Lớp:K58KTPM

Giảng viên: Đỗ Duy Cốp

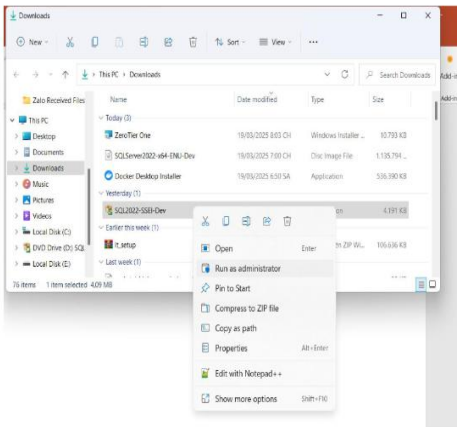
I. Cài đặt SQL Server 2022 Dev

Tải xuống các file cài cần thiết
B1: Vào đường dẫn sau:
https://www.microsoft.com/en-us/sql-server/sql-server-downloads
B2: Chọn bản **Developer**
B3: Tiếp tục tải xuống:
https://learn.microsoft.com/en-us/ssms/download-sql-server-management-studio-ssms
Chọn bản SSMS 20.2 mới nhất

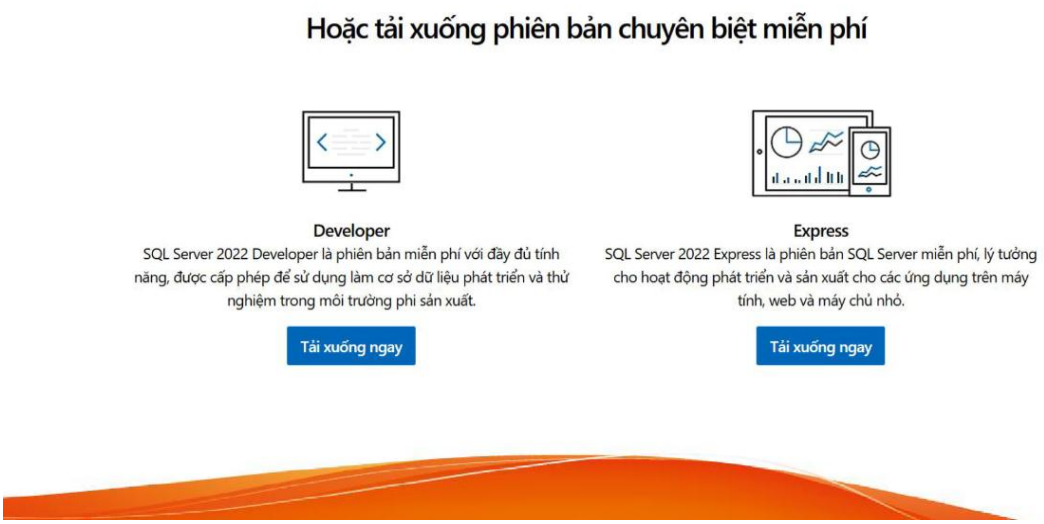


I. Cài đặt SQL Server 2022 Dev

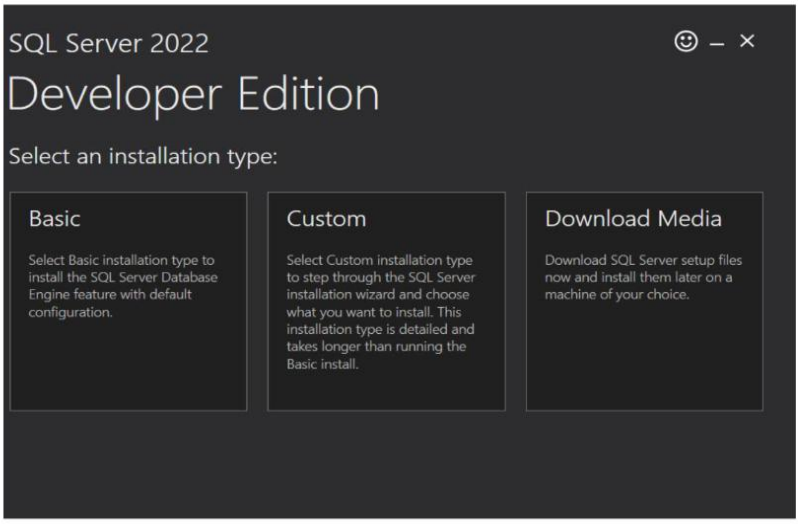
Tải xuống các file cài cần thiết
B1: Vào đường dẫn sau:
https://www.microsoft.com/en-us/sql-server/sql-server-downloads
B2: Chọn bản **Developer**
B3: Tiếp tục tải xuống:
https://learn.microsoft.com/en-us/ssms/download-sql-server-management-studio-ssms
Chọn bản SSMS 20.2 mới nhất



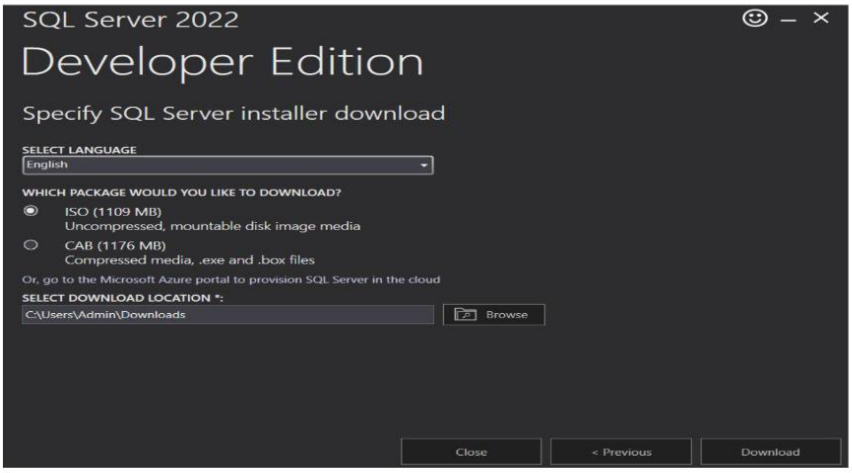
B4:Chọn Developer dể dowload



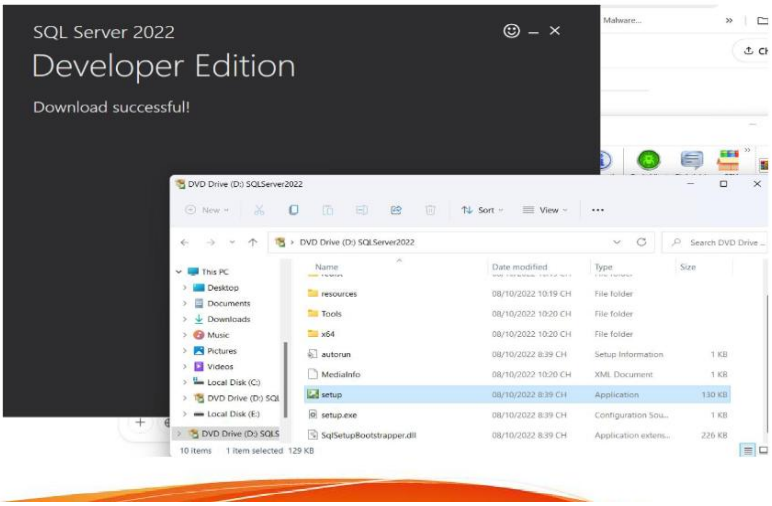
B5:Chọn download Media.



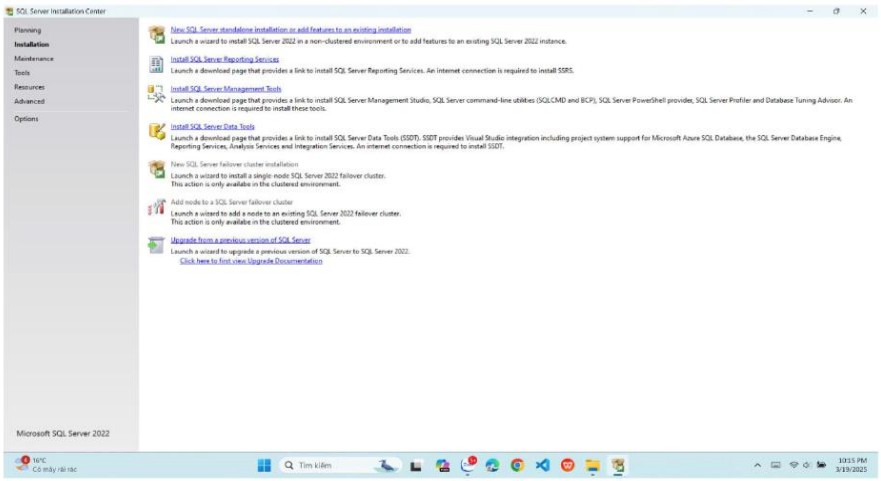
B6:Chọn ISO=> download nó xuống thành file iso bản dev.



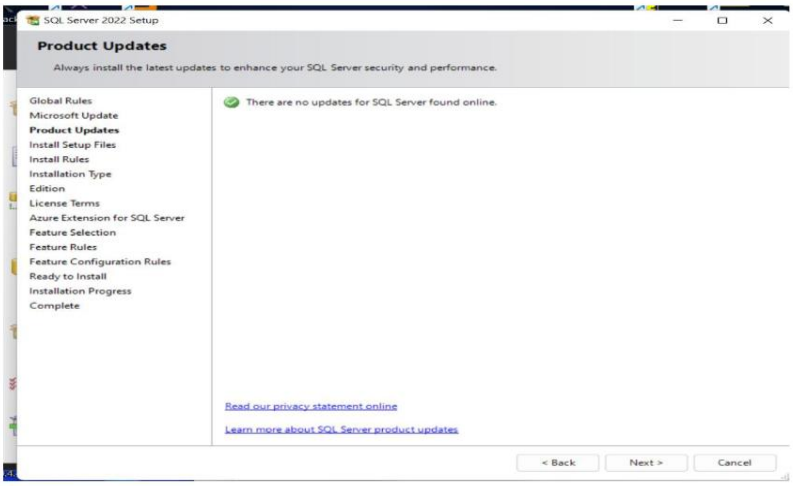
B7:Chạy file setup.exe như hình



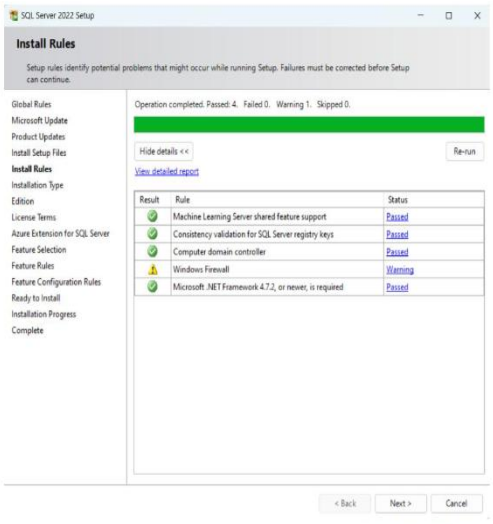
B8:Chọn New SQL Server standalone installation... Chọn Next

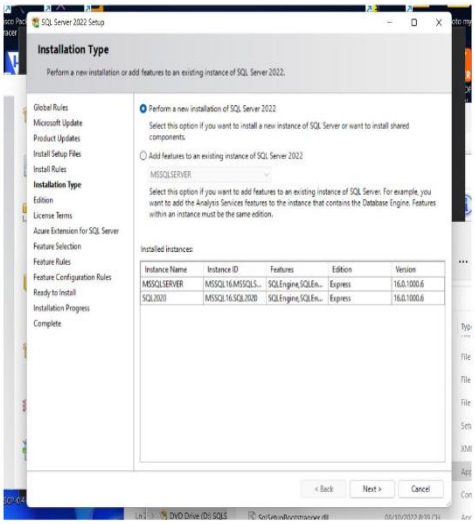


B9: SQL Server sẽ kiểm tra xem máy có đủ điều kiện cài đặt. Sau đó tiếp tục ấn Next

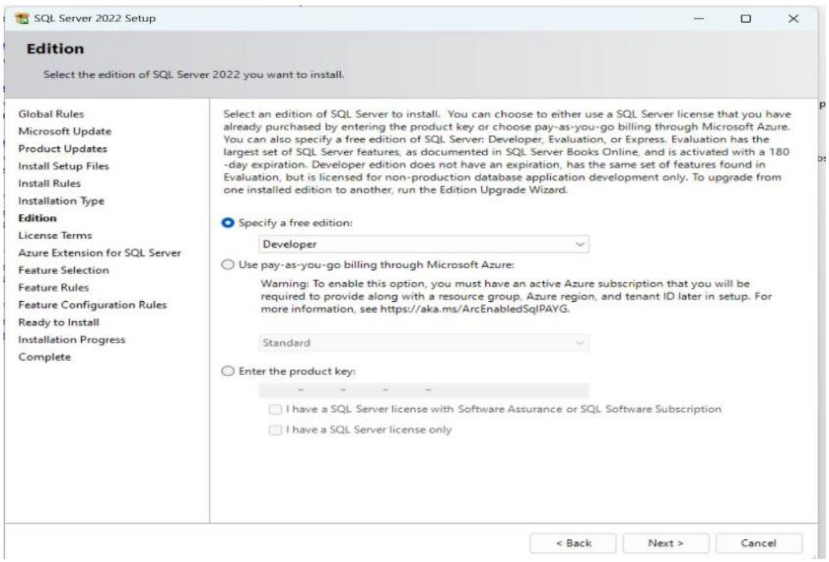


B10: Chọn Next

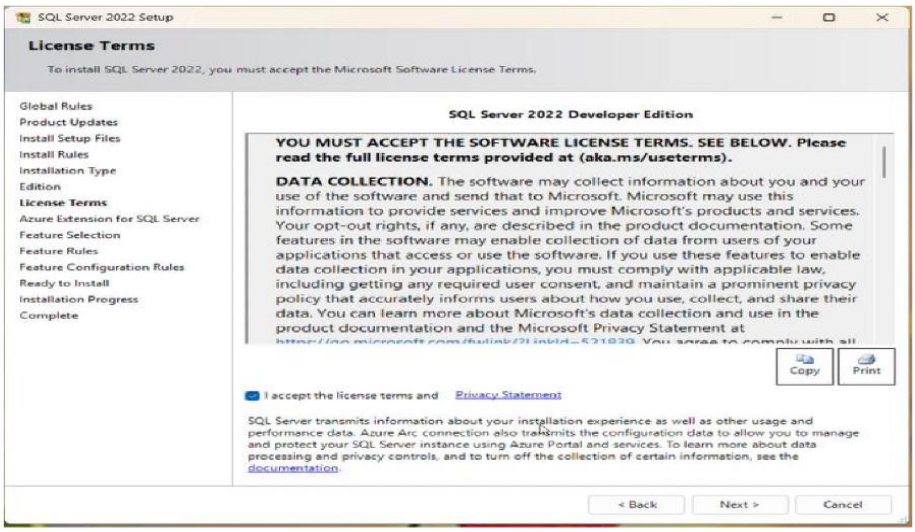




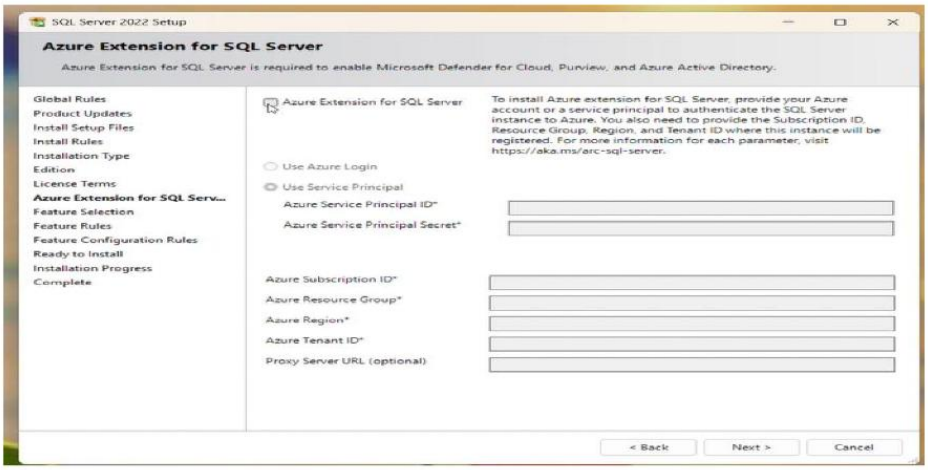
B11: Vì đang cài bản Dev, nên ta chọn **Developer** > Next



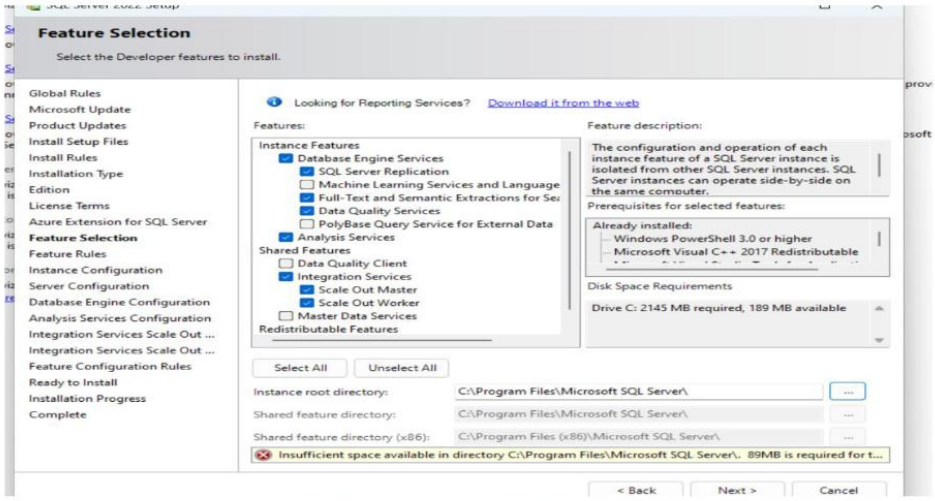
B12:Tích vào ô I accept the lincense tems and rồi ấn Next



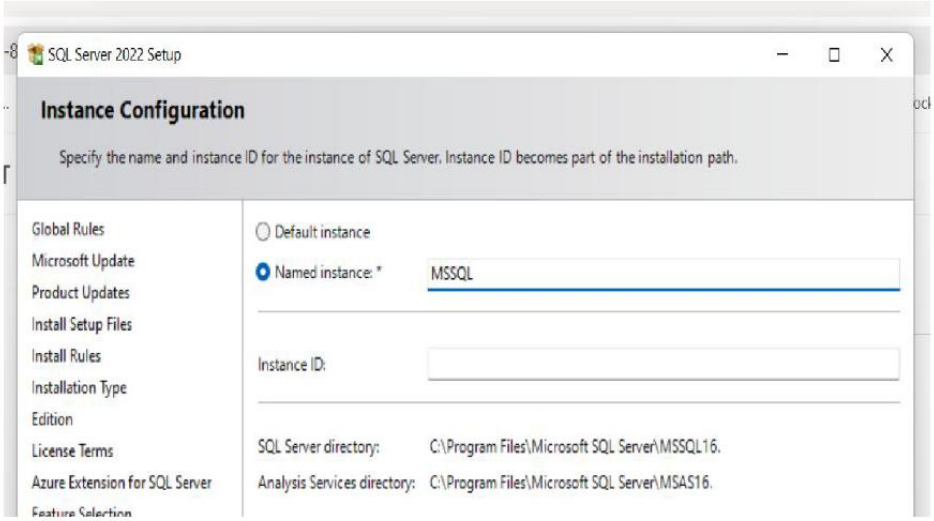
B13:Ở phần này thì bỏ tích chỗ Azure Extension for SQL Server rồi ấn next



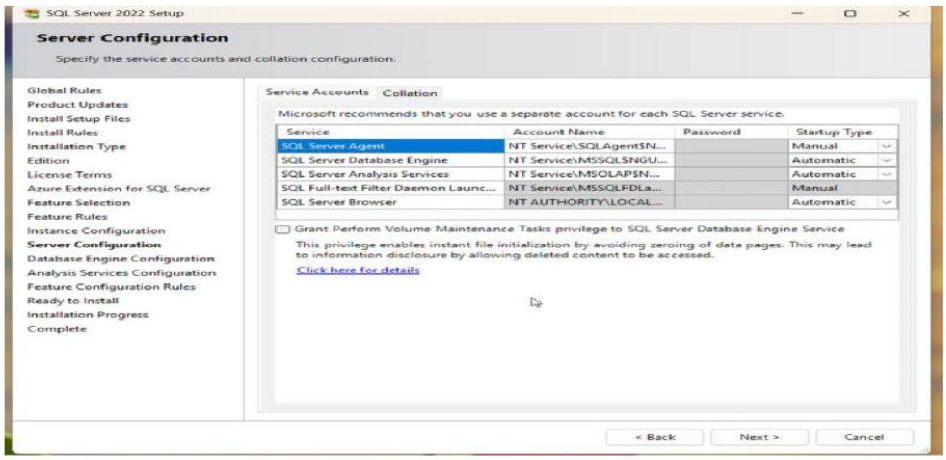
B14:Nhan select all > phần Instance Features>Rồi ấn Next



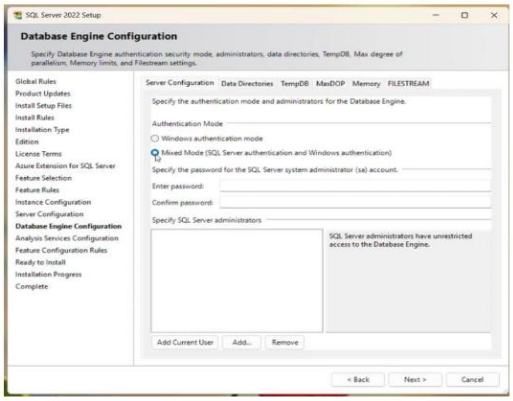
B15:Đặt tên SQL Server Instance, ở đây mình đặt là MSSQL



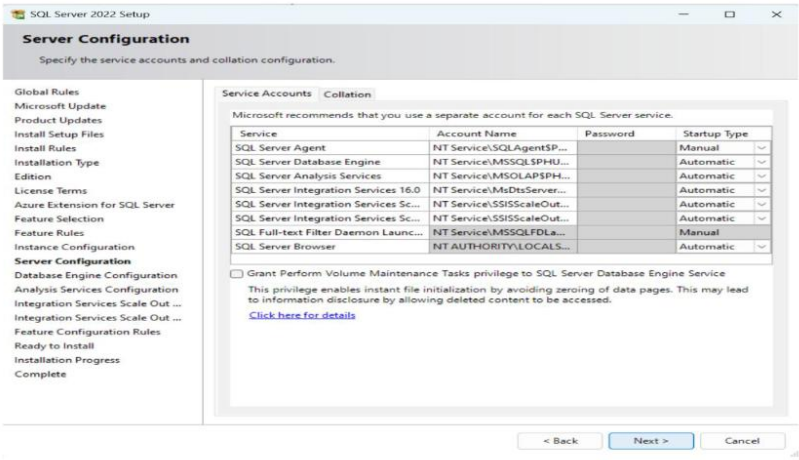
B16:Nhấn Next



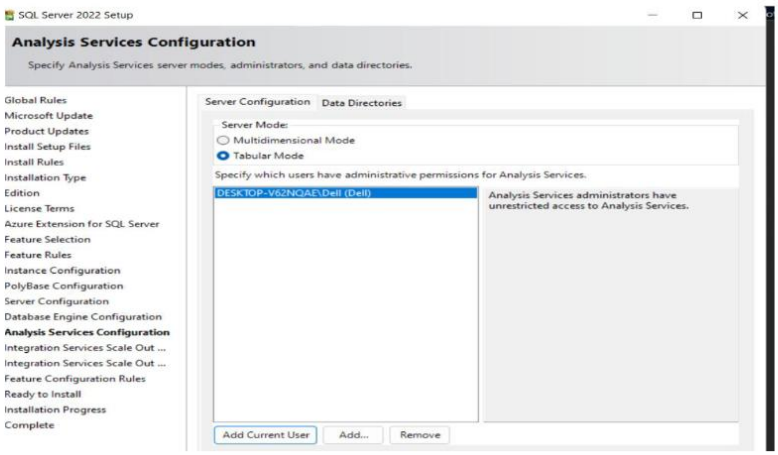
B17:Ở phần này là quan trọng nhất và nó có hai sự lựa chọn đó là Windows authentication mode hoặc Mixed Mode(SQL Server authentication and Windowns authentication).
Nếu chọn cái đầu tiên thì nó chỉ cho phép đăng nhập bằng tài khoản Windows của hệ thống, an toàn hơn vì quản lý quyền trên Windows,
còn cái Mixed Mode thì nó hỗ trợ cả xác thức Windowns và SQL Server(đăng nhập bằng tài khoản sa và mật khẩu) sau đó ấn Add Current User rồi ấn Next



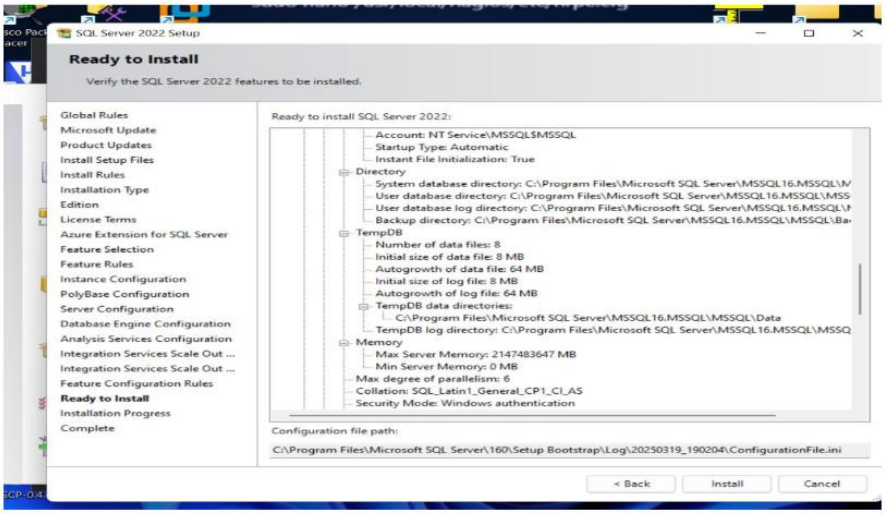
B16: Cấu hình Service cho SQL Server (Server Configuration). Nếu không có yêu cầu gì phức tạp, hãy giữ nguyên > Next



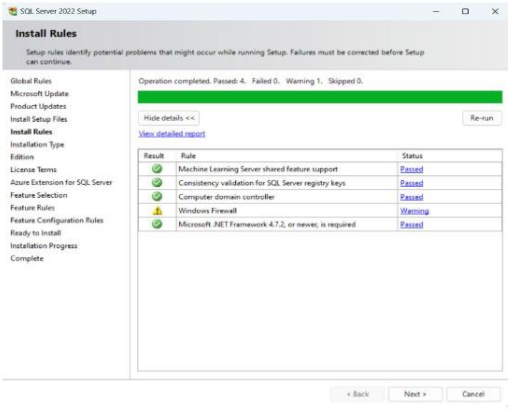
B17: Cấu hình Analysis Services, tại Server Mode (Chế độ máy chủ) hãy chọn Tabular Mode nếu bạn có nhu cầu vừa phải không chuyên sâu > Next

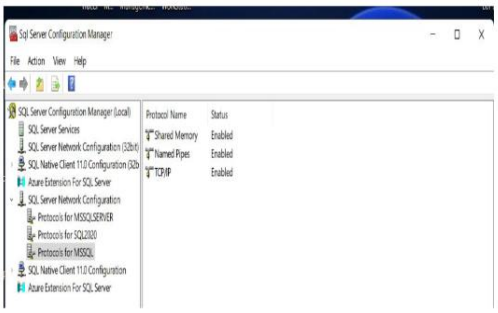


B18: Chọn Install và chờ cho đến khi chạy xong



B19: Sơ bộ thì đã cài thành công, thoát ra và truy cập Sql Server Configuration Manager, tại mục Server Network Configuration chọn Protocols for MSSQL (đây là tên ban đầu đã đặt ở B15). Hãy Enable toàn bộ Protocol name của nó.

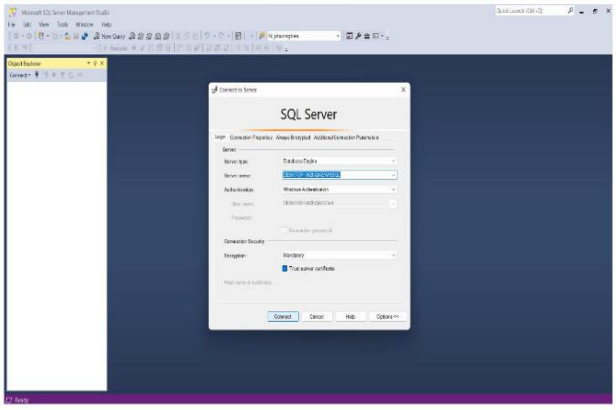




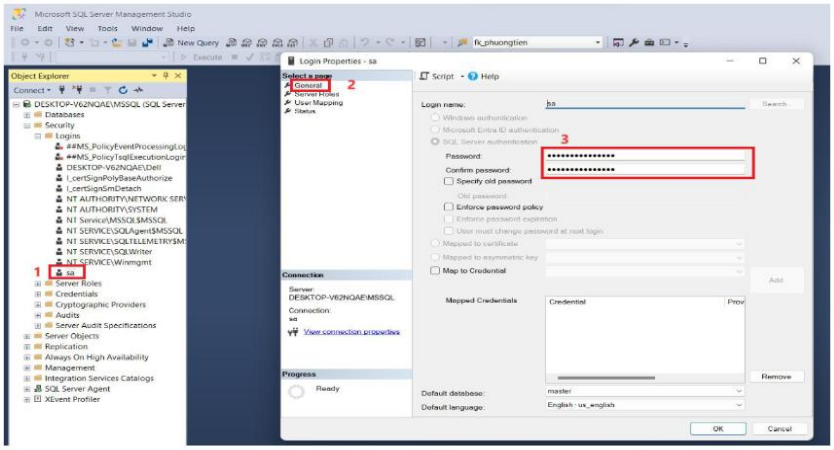
II. Đăng nhập từ sql managerment studio

Chạy file cài SSMS-Setup-ENU để cài SQL Server Management Studio 20 trước đó. Sau khi cài xong:

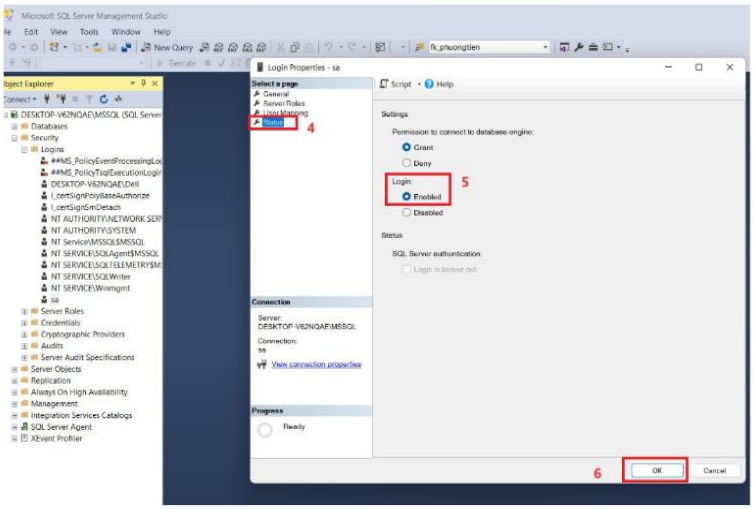
B1: Mở SQL Server Management Studio lên và gõ tên Server Name, tại Authentication:
1. Đăng nhập theo cách Windows Authentication, vì đang sử dụng Windows nên Connect



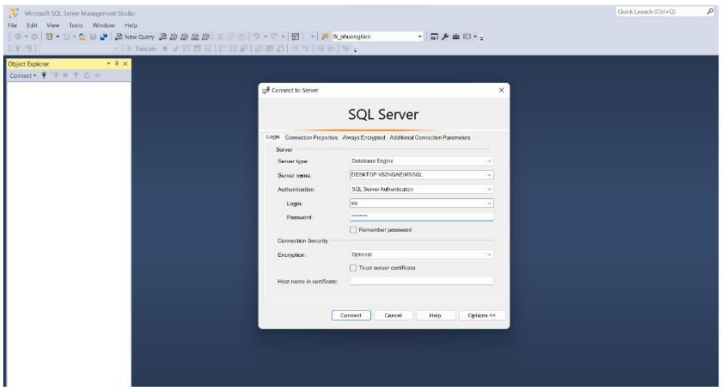
2.2. Đăng nhập theo cách Sql Server Auth:
Sau khi vào được rồi, Chọn Sencurty > Login > sa > General > đặt password dễ nhớ



Tại tab Status chọn Enable trong mục Login > ấn OK
Nhớ chuột phải Database > Restart để cập nhật thay đổi



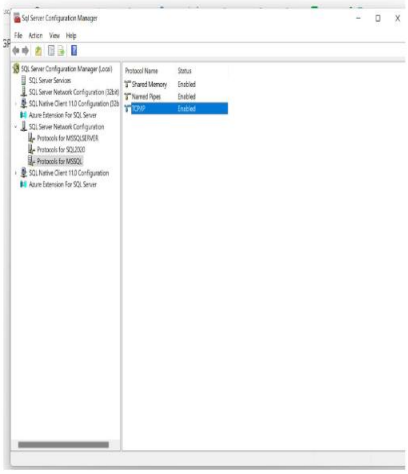
Đăng nhập



III. Cấu hình TCP/IP

B1: Mở SQL Sever Configuration Manager, chọn SQL Server Network Configuration,
Click vào Protocol for <Tên_đã_cài_trước đó>

B2: Chọn TCP/IP mở lên



3. Mã QR Github của bài tập

❖ Link và QR:
https://github.com/XuanPhuong01/bai-tap

CHƯƠNG 2. BÀI TẬP VỀ NHÀ SỐ 2

1. Đề Bài

BÀI TẬP VỀ NHÀ 02 - MÔN HỆ QUẢN TRỊ CSDL:
DEADLINE: 23H59 NGÀY 25/03/2025
ĐIỀU KIỆN: (ĐÃ LÀM XONG BÀI 1)

1. Đã cài đặt SQL Server 2022 Dev.
2. Đã cài đặt SQL Managerment Studio bản mới nhất.
3. Đã kết nối từ SQL Managerment Studio vào SQL Server.
4. Đã có tài khoản github, biết cách tạo repository(kho lưu trữ) cho phép truy cập public.

BÀI TOÁN:

- Tạo csdl quan hệ với tên QLSV gồm các bảng sau:
    - SinhVien(#masv,hoten,NgaySinh)
    - Lop(#maLop,tenLop)
    - GVCN(#@maLop,#@magv,#HK)
    - LopSV(#@maLop,#@maSV,ChucVu)
    - GiaoVien(#magv,hoten,NgaySinh,@maBM)
    - BoMon(#MaBM,tenBM,@maKhoa)
    - Khoa(#maKhoa,tenKhoa)
    - MonHoc(#mamon,Tenmon,STC)
    - LopHP(#maLopHP,TenLopHP,HK,@maMon,@maGV)
    - DKMH(#@maLopHP,#@maSV,DiemTP,DiemThi,PhanTramThi)

YÊU CẦU:

1. Thực hiện các hành động sau trên giao diện đồ hoạ để tạo cơ sở dữ liệu cho bài toán:

- Tạo database mới, mô tả các tham số(nếu có) trong quá trình.
- Tạo các bảng dữ liệu với các trường như mô tả, chọn kiểu dữ liệu phù hợp với thực tế (tự tìm hiểu)
- Mỗi bảng cần thiết lập PK, FK(s) và CK(s) nếu cần thiết. (chú ý dấu # và @: # là chỉ PK, @ chỉ FK)

2. Chuyển các thao tác đồ hoạ trên thành lệnh SQL tương đương. lưu tất cả các lệnh SQL trong file: Script_DML.sql

HÌNH THỨC LÀM BÀI:

1. Tạo repository mới, tạo file readme.md (có hướng dẫn trên zalo group)
2. Sinh viên thao tác trên máy tính cá nhân, chụp màn hình quá trình làm, chỉ cần chụp active window, thi thoảng chụp full màn hình để thấy sự cá nhân hoá.
3. Hình sau khi chụp paste trực tiếp vào file readme trên github, cần mô tả các phần trên ảnh để tỏ ra là hiểu hết!
4. upload các file liên quan: Script_DML.sql
5. Update link của repository vào cột bài tập 2 trên file excel online của thầy (đã ghim link trên zalo group)

Chú ý:

1. Được phép dùng AI và tham khảo bài của bạn, nhưng phải có sự khác biệt đáng kể.

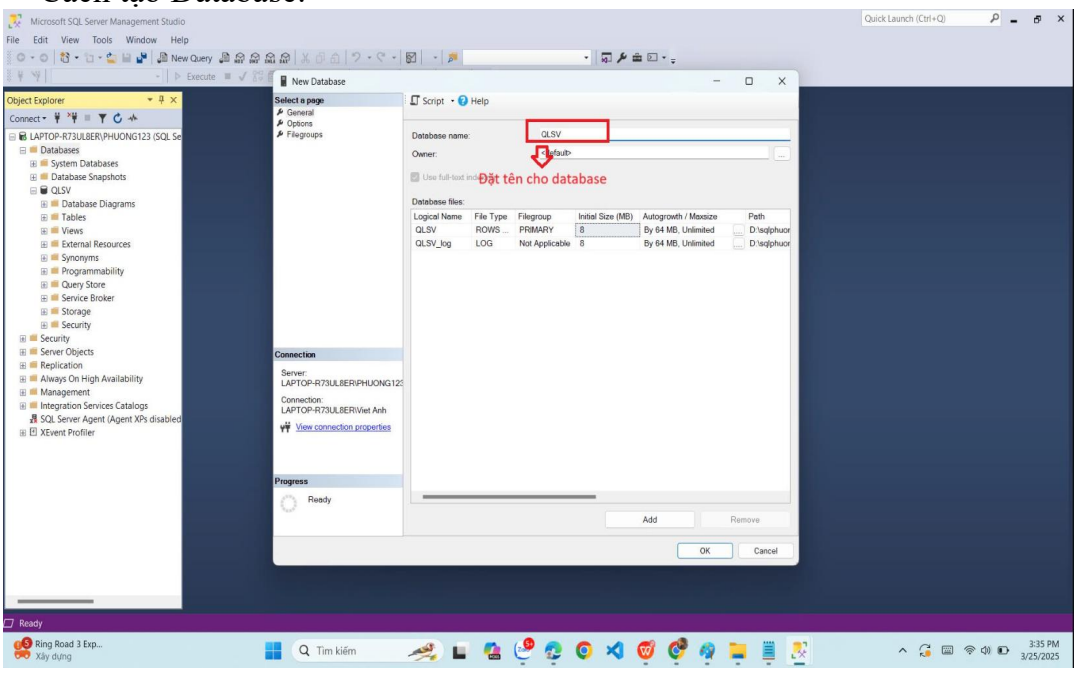
2. Nghiêm cấm copy, clone. Tham khảo và copy là 2 việc khác hẳn nhau. Thầy có tool để check!
3. Bài làm phải có dấu ấn cá nhân (hãy sáng tạo và biết cách bảo vệ mình nếu bạn là bản chính)
4. Kết quả AI phải phù hợp với yêu cầu, nếu quá sai lệch <=> sv ko đọc => Cấm thi
5. Nên nhớ: cấm thi là ko có vùng cấm và thầy chưa bao giờ nói đùa về việc cấm thi.

2. *Bài Làm*

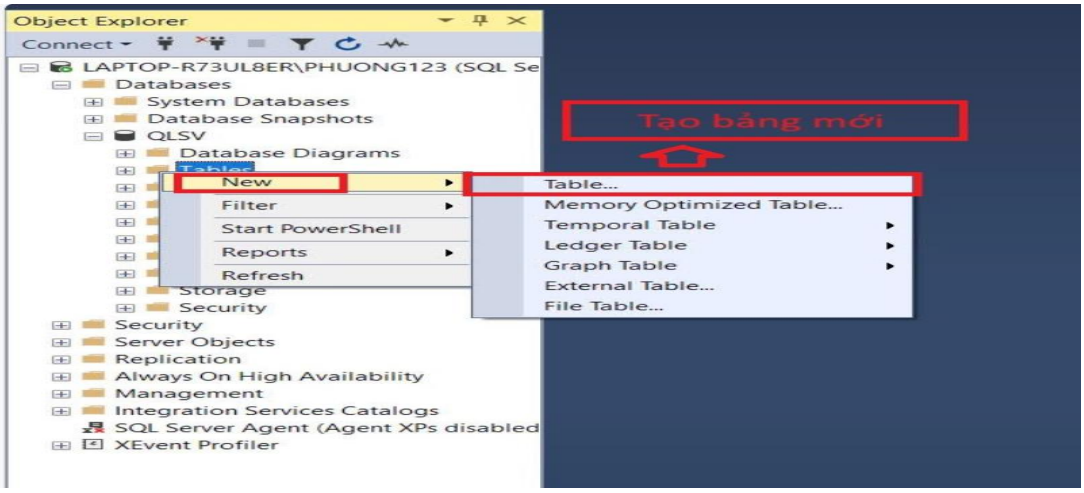
2.1. Các bước

❖ *Mô tả việt tạo Database*

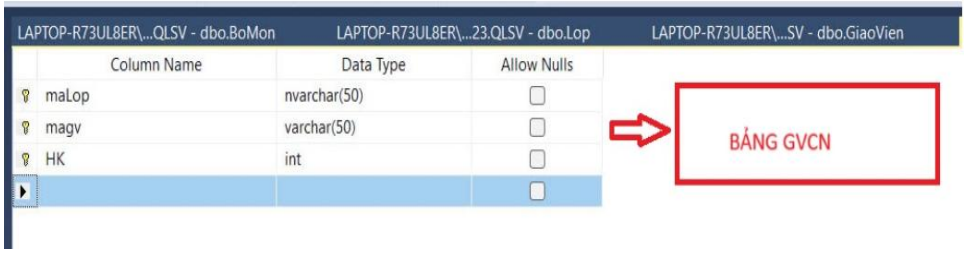
- Cách tạo Database.



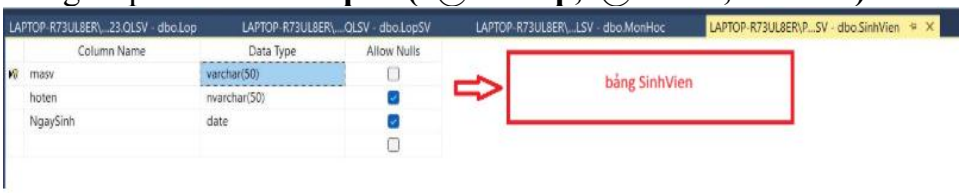
- Cách Tạo Bảng:



❖ Tạo bảng GVCN **GVCN(#@maLop,#@magv,#HK)**



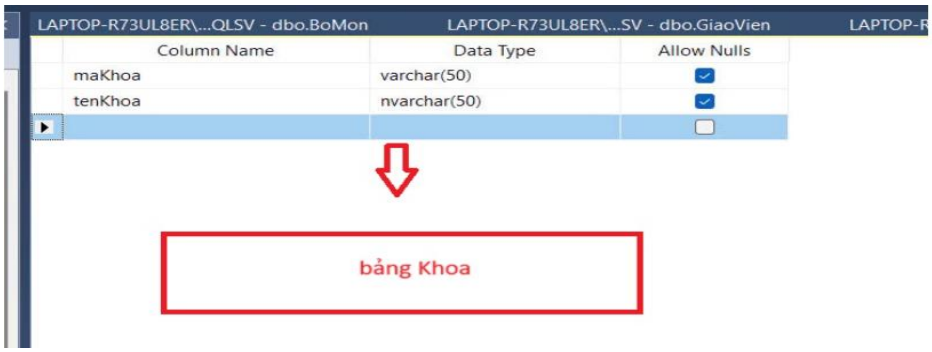
❖ Tạo Bảng Lớp Sinh Viên **LopSV(#@maLop,#@maSV,ChucVu)**



❖ Tạo Bảng Giáo Viên **GiaoVien(#magv,hoten,NgaySinh,@maBM)**

❖ Tạo Bảng Bộ Môn **BoMon(#MaBM,tenBM,@maKhoa)**

❖ Tạo Bảng Khoa **Khoa(#maKhoa,tenKhoa)**



❖ Tạo Bảng Môn Học **MonHoc(mamon,Tenmon,STC)**

❖ Tạo Bảng DKMH **DKMH (#@maLopHP, #@maSV, DiemTP, DiemThi, PhanTramThi)**

2.2 .Code

❖ Code bảng sinh viên

```
CREATE TABLE [dbo].[sinhvien](
    [masv] [nvarchar](50) NOT NULL,
    [hoten] [nvarchar](50) NULL,
    [ngaysinh] [date] NULL,
 CONSTRAINT [PK_sinhvien] PRIMARY KEY CLUSTERED
(
    [masv] ASC
))
GO
```

❖ Code bảng môn học

```
CREATE TABLE [dbo].[monhoc](
    [mamon] [nvarchar](50) NOT NULL,
    [tenmon] [nvarchar](50) NULL,
    [stc] [nvarchar](15) NULL,
 CONSTRAINT [PK_monhoc] PRIMARY KEY CLUSTERED
(
    [mamon] ASC
))
GO
```

❖ Code bảng lớp sinh viên

```
CREATE TABLE [dbo].[lopsv](
    [malop] [nvarchar](10) NOT NULL,
    [masv] [nvarchar](15) NOT NULL,
    [chucvu] [nvarchar](50) NULL,
 CONSTRAINT [PK_lopsv] PRIMARY KEY CLUSTERED
(
    [malop] ASC,
```

```
    [masv] ASC
))
GO
```

❖ Code bảng lớp

```
CREATE TABLE [dbo].[lop](
    [malop] [nvarchar](10) NOT NULL,
    [tenlop] [nvarchar](50) NULL,
 CONSTRAINT [PK_lop] PRIMARY KEY CLUSTERED
(
    [malop] ASC
))
GO
```

❖ Code bảng lớp học phần

```
CREATE TABLE [dbo].[lopHP1](
    [malopHP] [nvarchar](15) NOT NULL,
    [tenlopHP] [nvarchar](50) NULL,
    [hk] [nvarchar](50) NULL,
    [mamon] [nvarchar](15) NULL,
    [magv] [nvarchar](15) NULL,
 CONSTRAINT [PK_lopHP1] PRIMARY KEY CLUSTERED
(
    [malopHP] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
ALTER TABLE [dbo].[lopHP1]  WITH CHECK ADD  CONSTRAINT
[FK_lopHP1_giaovien] FOREIGN KEY([magv])
REFERENCES [dbo].[giaovien] ([magv])
GO
ALTER TABLE [dbo].[lopHP1] CHECK CONSTRAINT
[FK_lopHP1_giaovien]
GO
ALTER TABLE [dbo].[lopHP1]  WITH CHECK ADD  CONSTRAINT
[FK_lopHP1_monhoc] FOREIGN KEY([mamon])
REFERENCES [dbo].[monhoc] ([mamon])
GO
ALTER TABLE [dbo].[lopHP1] CHECK CONSTRAINT
[FK_lopHP1_monhoc]
GO
```

❖ Code bảng khoa

```
CREATE TABLE [dbo].[khoa](
    [makhoa] [nvarchar](13) NOT NULL,
    [tenkhoa] [nchar](10) NULL,
 CONSTRAINT [PK_khoa] PRIMARY KEY CLUSTERED
(
    [makhoa] ASC
))
GO
```

❖ Code bảng giáo viên

```
CREATE TABLE [dbo].[giaovien](
    [magv] [nvarchar](15) NOT NULL,
    [hoten] [nvarchar](50) NULL,
    [ngaysinh] [date] NULL,
    [mabm] [nvarchar](13) NULL,
 CONSTRAINT [PK_giaovien] PRIMARY KEY CLUSTERED
(
    [magv] ASC
))
GO
ALTER TABLE [dbo].[giaovien]  WITH CHECK ADD  CONSTRAINT
[FK_giaovien_bomon] FOREIGN KEY([mabm])
REFERENCES [dbo].[bomon] ([mabm])
GO
ALTER TABLE [dbo].[giaovien] CHECK CONSTRAINT
[FK_giaovien_bomon]
GO
```

❖ Code bảng GVCN

```
CREATE TABLE [dbo].[gvcn](
    [malop] [nvarchar](10) NOT NULL,
    [magv] [nvarchar](15) NOT NULL,
    [hk] [nvarchar](50) NOT NULL,
 CONSTRAINT [PK_gvcn] PRIMARY KEY CLUSTERED
(
    [malop] ASC,
    [magv] ASC,
    [hk] ASC
))
GO
ALTER TABLE [dbo].[gvcn]  WITH CHECK ADD  CONSTRAINT
[FK_gvcn_giaovien] FOREIGN KEY([magv])
REFERENCES [dbo].[giaovien] ([magv])
GO
ALTER TABLE [dbo].[gvcn] CHECK CONSTRAINT [FK_gvcn_giaovien]
GO
ALTER TABLE [dbo].[gvcn]  WITH CHECK ADD  CONSTRAINT
[FK_gvcn_lop] FOREIGN KEY([malop])
```

```
REFERENCES [dbo].[lop] ([malop])
GO
ALTER TABLE [dbo].[gvcn] CHECK CONSTRAINT [FK_gvcn_lop]
GO
```

❖ Code bảng DKMH

```
CREATE TABLE [dbo].[DKMH](
    [malopHP] [nvarchar](15) NOT NULL,
    [masv] [nvarchar](15) NULL,
    [diemTP] [int] NULL,
    [diemthi] [int] NULL,
    [phantramthi] [int] NULL,
 CONSTRAINT [PK_DKMH] PRIMARY KEY CLUSTERED
(
    [malopHP] ASC
))
GO
ALTER TABLE [dbo].[DKMH]  WITH CHECK ADD  CONSTRAINT
[FK_DKMH_lopHP] FOREIGN KEY([malopHP])
REFERENCES [dbo].[lopHP] ([malopHP])
GO
ALTER TABLE [dbo].[DKMH] CHECK CONSTRAINT [FK_DKMH_lopHP]
GO
ALTER TABLE [dbo].[DKMH]  WITH CHECK ADD  CONSTRAINT
[FK_DKMH_sinhvien] FOREIGN KEY([masv])
REFERENCES [dbo].[sinhvien] ([masv])
GO
ALTER TABLE [dbo].[DKMH] CHECK CONSTRAINT
[FK_DKMH_sinhvien]
GO
```

❖ Code bảng bộ môn

```
CREATE TABLE [dbo].[bomon](
    [mabm] [varchar](50) NOT NULL,
    [tenbm] [varchar](50) NULL,
    [makhoa] [varchar](50) NULL,
 CONSTRAINT [PK_bomon] PRIMARY KEY CLUSTERED
(
    [mabm] ASC
))
GO

ALTER TABLE [dbo].[bomon]  WITH CHECK ADD  CONSTRAINT
[FK_bomon_khoa] FOREIGN KEY([makhoa])
REFERENCES [dbo].[khoa] ([makhoa])
GO
ALTER TABLE [dbo].[bomon] CHECK CONSTRAINT [FK_bomon_khoa]
GO
```

3. Mã QR của bài

❖ Link và QR:
https://github.com/XuanPhuong01/58KMT.git

CHƯƠNG 3. BÀI TẬP VỀ NHÀ 3

1. Đề bài

BÀI TẬP VỀ NHÀ 03 - MÔN HỆ QUẢN TRỊ CSDL:
DEADLINE: 23H59 NGÀY 30/03/2025
ĐIỀU KIỆN: (ĐÃ LÀM XONG BÀI 2)
BÀI TOÁN: Sửa bài 2 để có csdl như sau:

- SinhVien(#masv,hoten,NgaySinh)
- Lop(#maLop,tenLop)
- GVCN(#@maLop,#@magv,#HK)
- LopSV(#@maLop,#@maSV,ChucVu)
- GiaoVien(#magv,hoten,NgaySinh,@maBM)
- BoMon(#MaBM,tenBM,@maKhoa)
- Khoa(#maKhoa,tenKhoa)
- MonHoc(#mamon,Tenmon,STC)
- LopHP(#maLopHP,TenLopHP,HK,@maMon,@maGV)
- DKMH(#id_dk, @maLopHP,@maSV,DiemThi,PhanTramThi)
- Diem(#id, @id_dk, diem)

YÊU CẦU:

1. Sửa bảng DKMH và bảng Điểm từ bài tập 2 để có các bảng như yêu cầu.
2. Nhập dữ liệu demo cho các bảng (nhập có kiểm soát từ tính năng Edit trên UI của mssm)
3. Viết lệnh truy vấn để: Tính được điểm thành phần của 1 sinh viên đang học tại 1 lớp học phần.

HÌNH THỨC LÀM BÀI:

1. Tạo file bai_tap3.md trên cùng repository của bài tập 2: Nội dung chứa đề bài, và ảnh chụp quá trình thao tác các yêu cầu khác.
2. Chụp ảnh quá trình sửa bảng DKMH và quá trình thêm bảng Diem, chú ý @ là FK, và thêm CK cho trường điểm
3. Hình sau khi chụp paste trực tiếp vào file bai_tap3.md trên github, cần mô tả các phần trên ảnh để tỏ ra là hiểu hết!
4. dùng tính năng: Tasks -> Generate Scrips => sinh ra file: bai_tap_3_schema.sql (chỉ chứa lệnh tạo cấu trúc của db)
5. dùng tính năng: Tasks -> Generate Scrips => advance => Check Data only => sinh ra file: bai_tap_3_data.sql (chỉ chứa dữ liệu đã nhập demo vào db)
6. Tạo diagram mô tả các PK, FK của db. Chụp hình kết quả các bảng có các đường nối 1-->nhiều
7. upload 2 file bai_tap_3_schema.sql và bai_tap_3_data.sql lên repository.
8. nhớ commit để save nội dung file bai_tap3.md

DEADLINE: 23H59 NGÀY 30/03/2025

Chú ý:

1. Được phép dùng AI và tham khảo bài của bạn, nhưng phải có sự khác biệt đáng kể.
2. Nghiêm cấm copy, clone. Tham khảo và copy là 2 việc khác hẳn nhau. Thầy có tool để check!

3. Bài làm phải có dấu ấn cá nhân (hãy sáng tạo và biết cách bảo vệ mình nếu bạn là bản chính)
4. Kết quả AI phải phù hợp với yêu cầu, nếu quá sai lệch <=> sv ko đọc => Cấm thi
5. Nên nhớ: cấm thi là ko có vùng cấm và thầy chưa bao giờ nói đùa về việc cấm thi.

Nhắc lại nội quy học tập: SV vi phạm 1 trong các lỗi sau chỉ 1 lần sẽ bị cấm thi:

1. Nghỉ ko lý do chính đáng.
2. Không làm bài tập về nhà.
3. Vắng bài kiểm tra.
4. Nói chuyện tự do trong lớp.

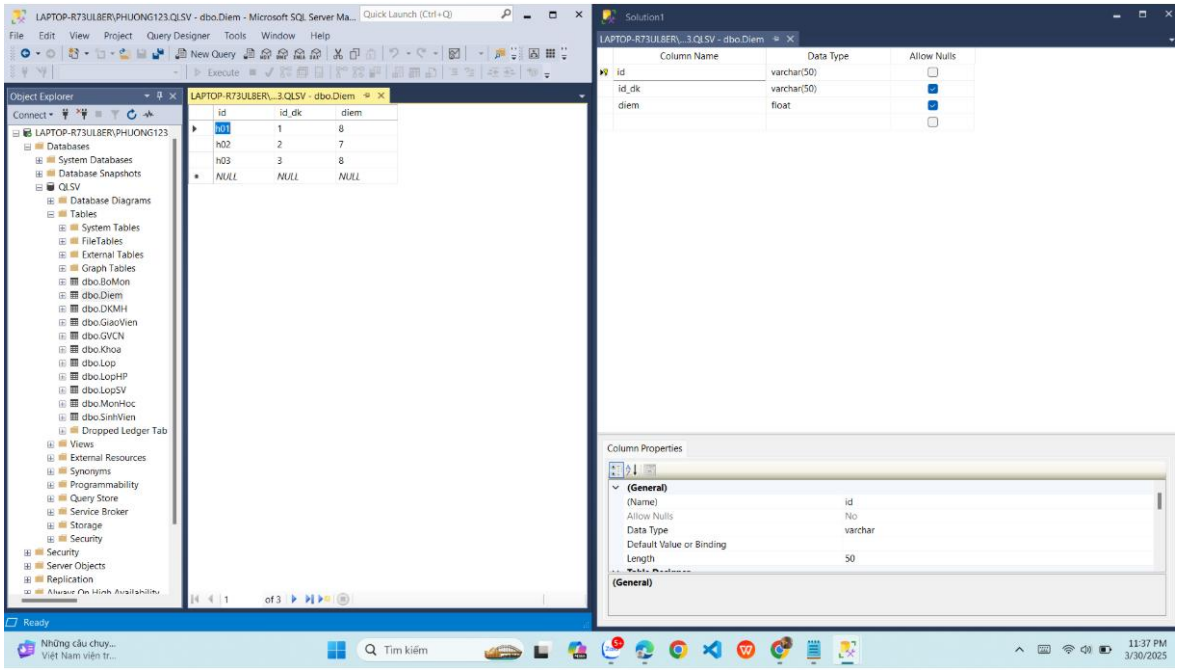
Bên cạnh đó, sẽ có điểm thưởng 10đ cho sv :

1. Trả lời đúng câu hỏi trên lớp.
2. Hỏi câu hỏi làm thầy khó trả lời.

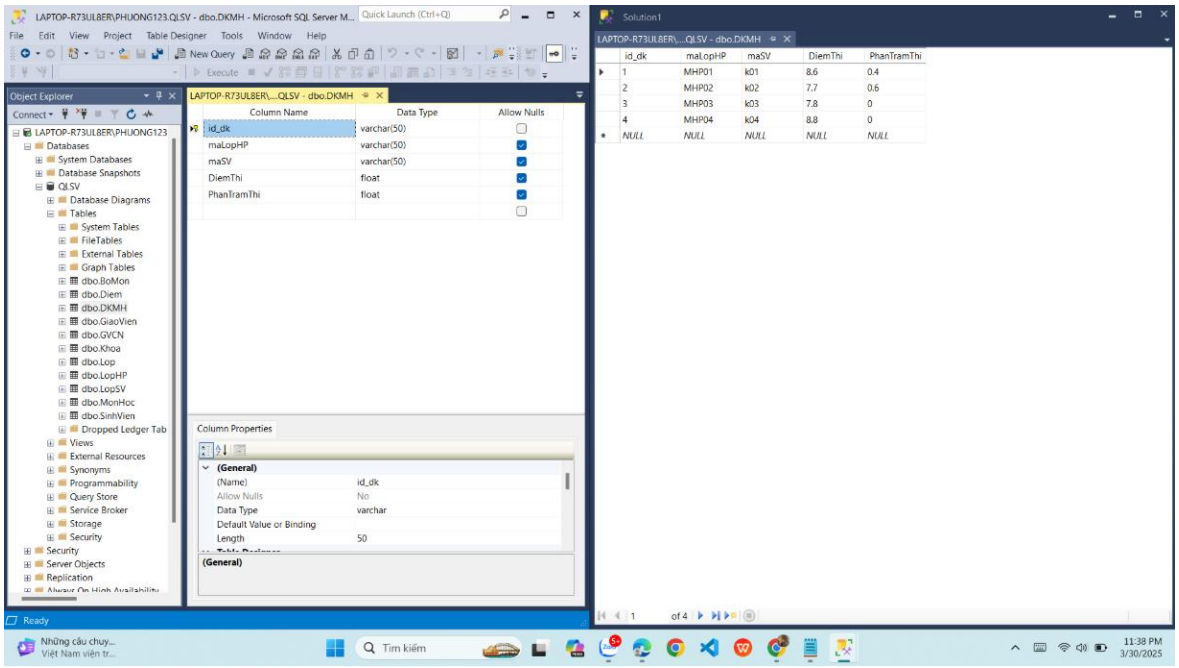
---NHẮC LẠI THỜI HẠN DEADLINE: 23H59 NGÀY 30/03/2025---

2. *Bài Làm*

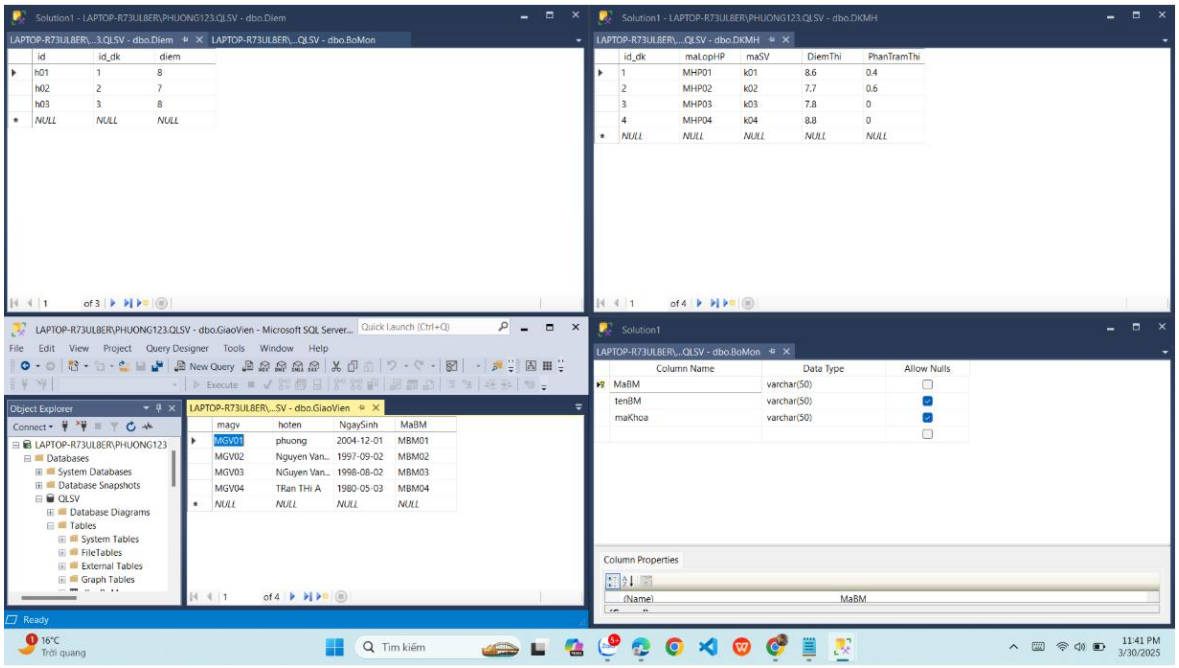
❖ Tạo Bảng Điểm



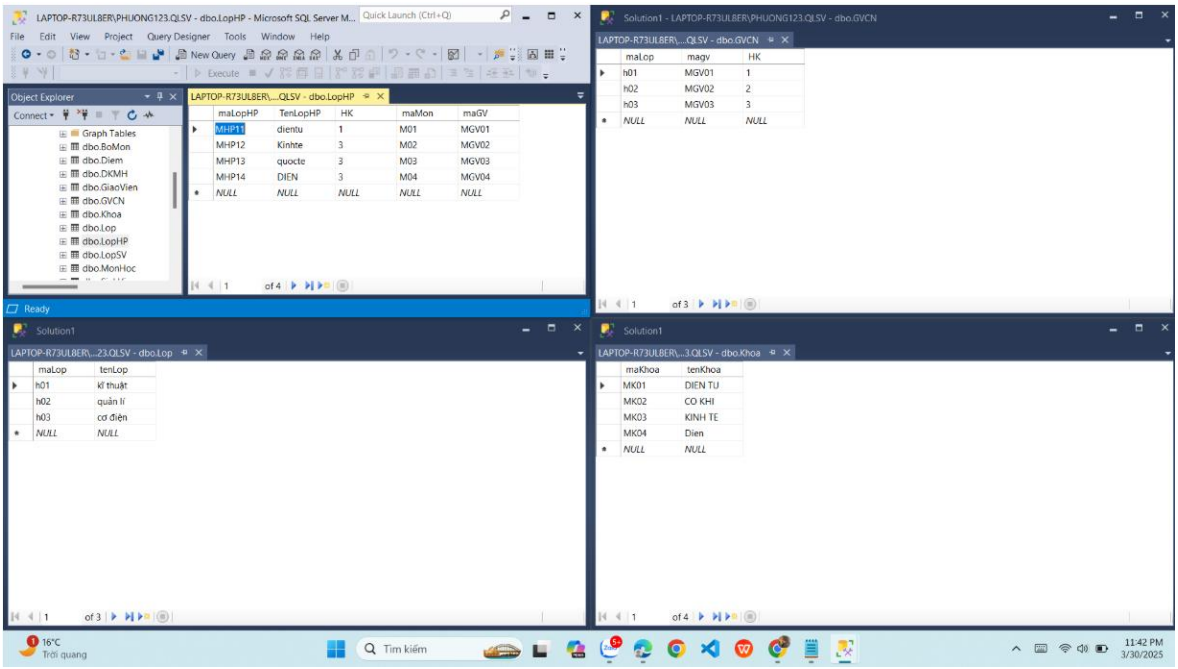
❖ Tạo bảng DKMH



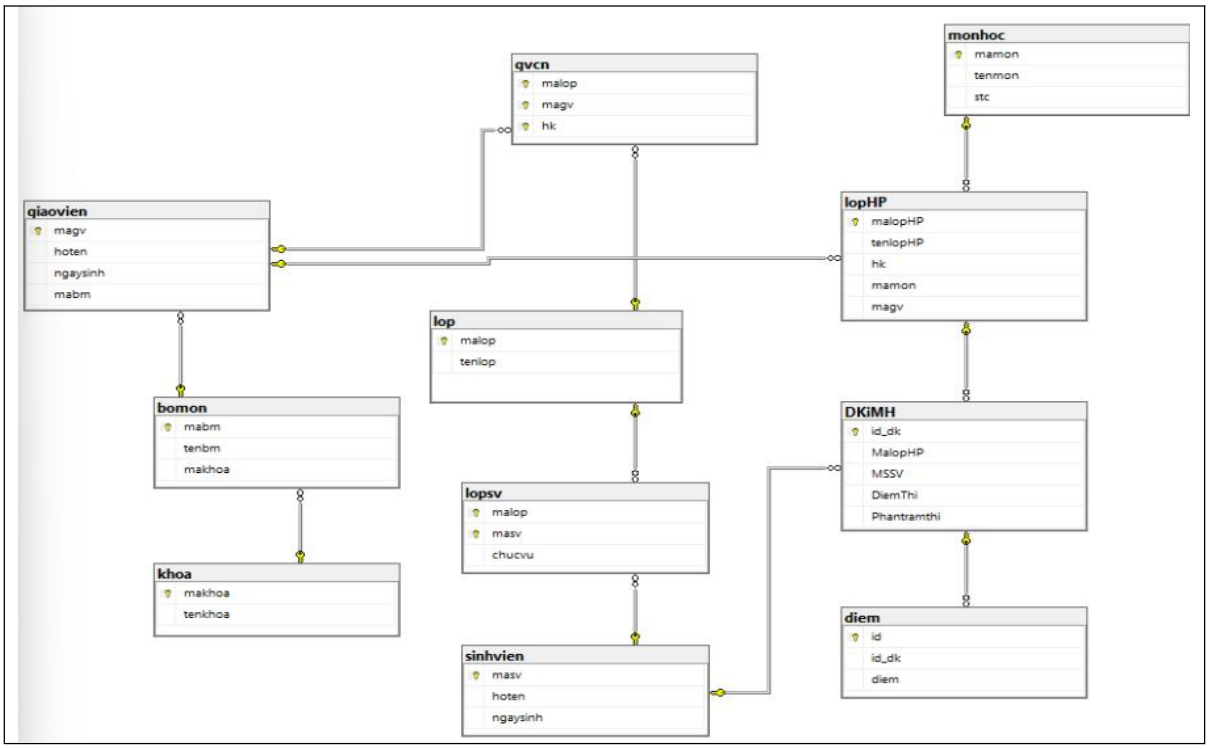
❖ Tạo bảng GiaoVien



❖ Tạo bảng LopHP



❖ Sơ Đồ Thực Thể



❖ Code truy vấn được lấy từ bảng sinh viên, lớp học phần, môn học(ở bài 2) bảng điểm và bảng DKMH

➢ Code truy vấn thành công

```
SELECT
   sv.masv,
   sv.hoten,
   lhp.TenLopHP,
   mh.tenmon,
   lhp.HK AS HocKy,
   SUM(d.diem) * (1 - CAST(dkmh.PhanTramThi AS FLOAT) / 100) AS
DiemThanhPhan
FROM SinhVien sv
JOIN DKiMH dkmh ON sv.masv = dkmh.mssv
JOIN LopHP lhp ON dkmh.maLopHP = lhp.maLopHP
JOIN monhoc mh ON lhp.mamon = mh.mamon
JOIN Diem d ON dkmh.id_dk = d.id_dk
GROUP BY sv.masv, sv.hoten, lhp.TenLopHP, mh.tenmon,
dkmh.PhanTramThi, lhp.HK;
```

➢ Dữ liệu đã nhập

```
   USE [csdl]
GO
INSERT [dbo].[monhoc] ([mamon], [tenmon], [stc]) VALUES (N'M01', N'giai
tich 2', N'03')
INSERT [dbo].[monhoc] ([mamon], [tenmon], [stc]) VALUES (N'M02', N'C++',
N'03')
INSERT [dbo].[monhoc] ([mamon], [tenmon], [stc]) VALUES (N'M03', N'toan',
N'01')
GO
INSERT [dbo].[khoa] ([makhoa], [tenkhoa]) VALUES (N'MK01', N'DienTu    ')
INSERT [dbo].[khoa] ([makhoa], [tenkhoa]) VALUES (N'MK02', N'Kinh Te   ')
INSERT [dbo].[khoa] ([makhoa], [tenkhoa]) VALUES (N'MK03', N'CO Khi    ')
GO
INSERT [dbo].[bomon] ([mabm], [tenbm], [makhoa]) VALUES (N'MBM11',
N'CNTT', N'MK01')
INSERT [dbo].[bomon] ([mabm], [tenbm], [makhoa]) VALUES (N'MBM12',
N'Kinh Te', N'MK02')
INSERT [dbo].[bomon] ([mabm], [tenbm], [makhoa]) VALUES (N'MBM13',
N'Co Khi', N'MK03')
GO
INSERT [dbo].[giaovien] ([magv], [hoten], [ngaysinh], [mabm]) VALUES
(N'GV001', N'Nguyen Van A', CAST(N'1990-09-09' AS Date), N'MBM11')
INSERT [dbo].[giaovien] ([magv], [hoten], [ngaysinh], [mabm]) VALUES
(N'GV002', N'Tran Thi B', CAST(N'1991-09-07' AS Date), N'MBM12')
INSERT [dbo].[giaovien] ([magv], [hoten], [ngaysinh], [mabm]) VALUES
```

```
(N'GV003', N'Nguyen Thi C', CAST(N'1994-03-09' AS Date), N'MBM13')
GO
INSERT [dbo].[lopHP] ([malopHP], [tenlopHP], [hk], [mamon], [magv]) VALUES (N'malop01', N'lop A', N'01', N'M01', N'GV001')
INSERT [dbo].[lopHP] ([malopHP], [tenlopHP], [hk], [mamon], [magv]) VALUES (N'malop02', N'lopB', N'3', N'M02', N'GV002')
INSERT [dbo].[lopHP] ([malopHP], [tenlopHP], [hk], [mamon], [magv]) VALUES (N'malop03', N'lopC', N'2', N'M03', N'GV003')
GO
INSERT [dbo].[sinhvien] ([masv], [hoten], [ngaysinh]) VALUES (N'MSV01 ', N'Nguyen Thi Linh', CAST(N'2004-09-30' AS Date))
INSERT [dbo].[sinhvien] ([masv], [hoten], [ngaysinh]) VALUES (N'MSV02', N'Hoang Xuan Trang', CAST(N'2004-07-05' AS Date))
INSERT [dbo].[sinhvien] ([masv], [hoten], [ngaysinh]) VALUES (N'MSV05', N'Tran Thi An', CAST(N'2000-02-02' AS Date))
GO
INSERT [dbo].[lop] ([malop], [tenlop]) VALUES (N'LOP01', N'DVT')
INSERT [dbo].[lop] ([malop], [tenlop]) VALUES (N'LOP02', N'ABC')
INSERT [dbo].[lop] ([malop], [tenlop]) VALUES (N'LOP03', N'AAA')
GO
INSERT [dbo].[lopsv] ([malop], [masv], [chucvu]) VALUES (N'lop01', N'MSV01', N'sinh vien')
INSERT [dbo].[lopsv] ([malop], [masv], [chucvu]) VALUES (N'lop01', N'MSV02', N'sinh vien')
GO
INSERT [dbo].[DKiMH] ([id_dk], [MalopHP], [MSSV], [DiemThi], [Phantramthi]) VALUES (1, N'malop01', N'MSV01', 9, 9)
INSERT [dbo].[DKiMH] ([id_dk], [MalopHP], [MSSV], [DiemThi], [Phantramthi]) VALUES (2, N'malop02', N'MSV02', 10, 10)
INSERT [dbo].[DKiMH] ([id_dk], [MalopHP], [MSSV], [DiemThi], [Phantramthi]) VALUES (3, N'malop01', N'MSV05', 10, 9)
GO
INSERT [dbo].[gvcn] ([malop], [magv], [hk]) VALUES (N'LOP01', N'GV001', N'3')
INSERT [dbo].[gvcn] ([malop], [magv], [hk]) VALUES (N'LOP02', N'GV002', N'3')
INSERT [dbo].[gvcn] ([malop], [magv], [hk]) VALUES (N'LOP03', N'GV003', N'3')
GO
INSERT [dbo].[diem] ([id], [id_dk], [diem]) VALUES (1, 1, 9)
INSERT [dbo].[diem] ([id], [id_dk], [diem]) VALUES (2, 2, 9)
INSERT [dbo].[diem] ([id], [id_dk], [diem]) VALUES (3, 3, 10)
GO
```

3. Mã QR của bài

❖ Link và QR:
https://github.com/XuanPhuong01/58KMT.git

CHƯƠNG 4. BÀI TẬP VỀ NHÀ 4

1. Đề bài

bai tap 4: (sql server) yêu cầu bài toán:

- Tạo csdl cho hệ thống TKB (đã nghe giảng, đã xem cách làm)
- Nguồn dữ liệu: TMS.tnut.edu.vn
- Tạo các bảng tuỳ ý (3nf)
- Tạo được query truy vấn ra thông tin gồm 4 cột: họ tên gv, môn dạy, giờ vào lớp, giờ ra. trả lời câu hỏi: trong khoảng thời gian từ datetime1 tới datetime2 thì có những gv nào đang bận giảng dạy.

các bước thực hiện:

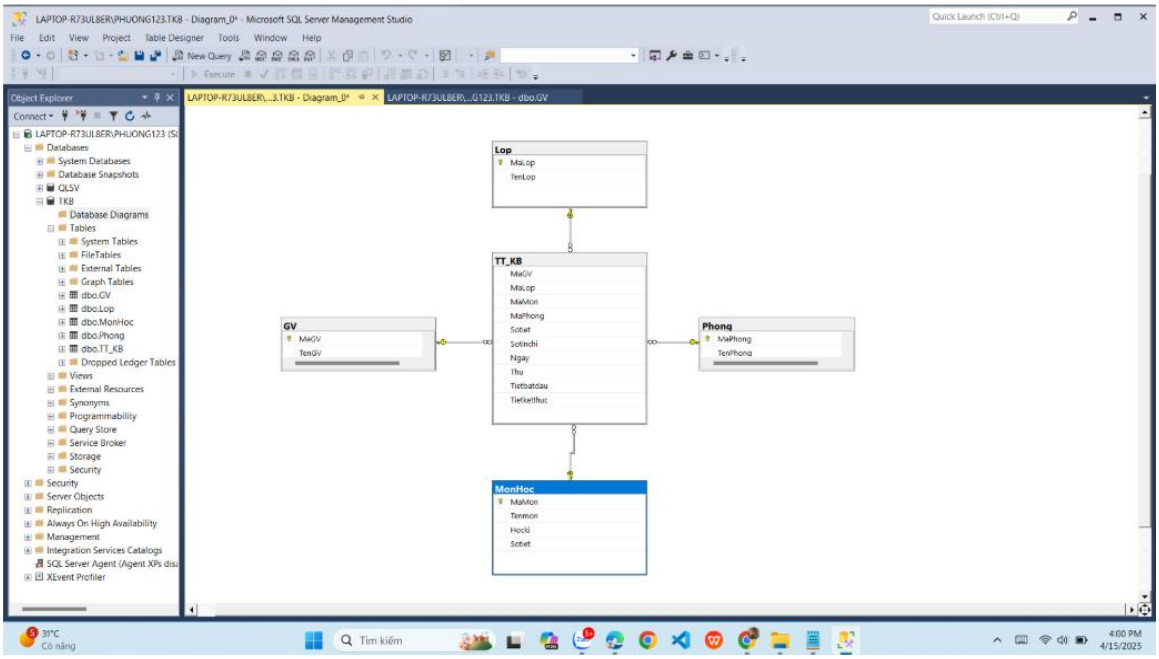
1. Tạo github repo mới: đặt tên tuỳ ý (có liên quan đến bài tập này)
2. tạo file readme.md, edit online nó: paste những ảnh chụp màn hình gõ text mô tả cho ảnh đó

Gợi ý: sử dung tms => dữ liệu thô => tiền xử lý => dữ liệu như ý (3nf) tạo các bảng với struct phù hợp insert nhiều rows từ excel vào cửa sổ edit dữ liệu 1 table (quan sát thì sẽ làm đc)

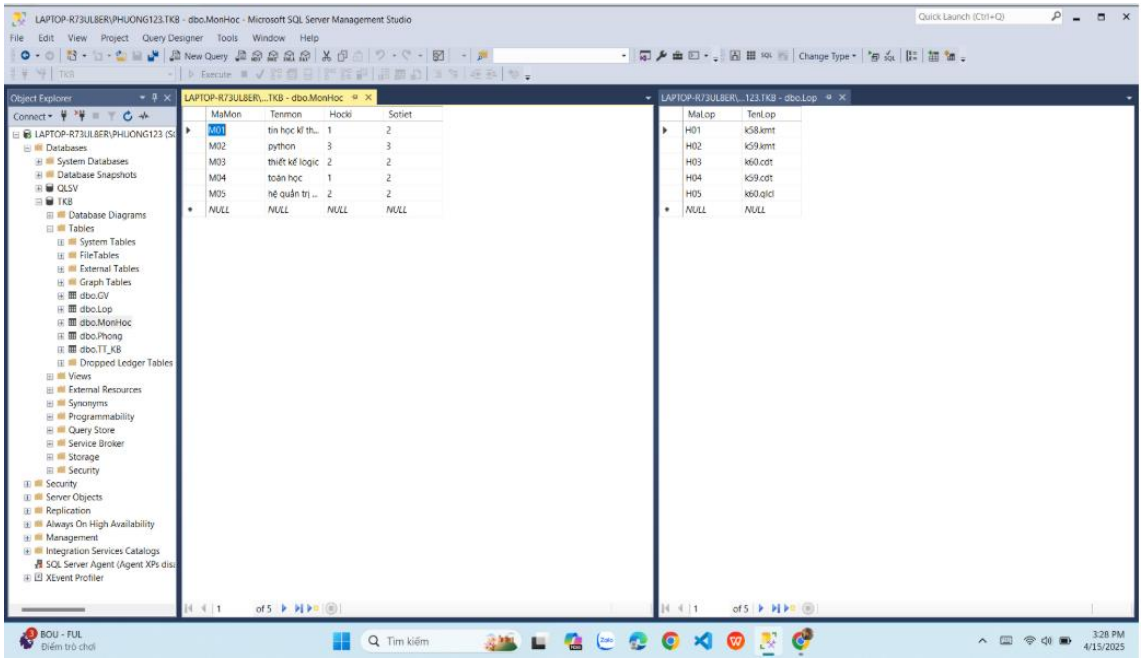
deadline: 15/4/2025

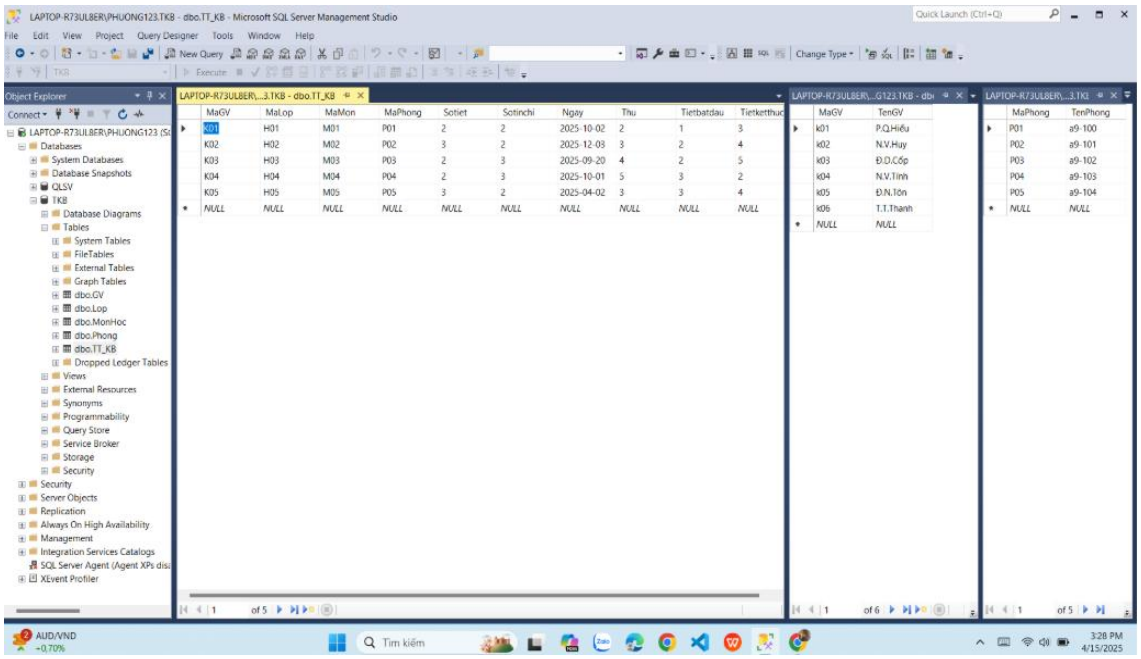
2. Bài làm

❖ B1: Tạo các bảng: GV,LOP,MONHOC,PHONG,TT_KB và chuẩn hóa các bảng dưới dạng 3NF ta được sơ đồ liên kết:

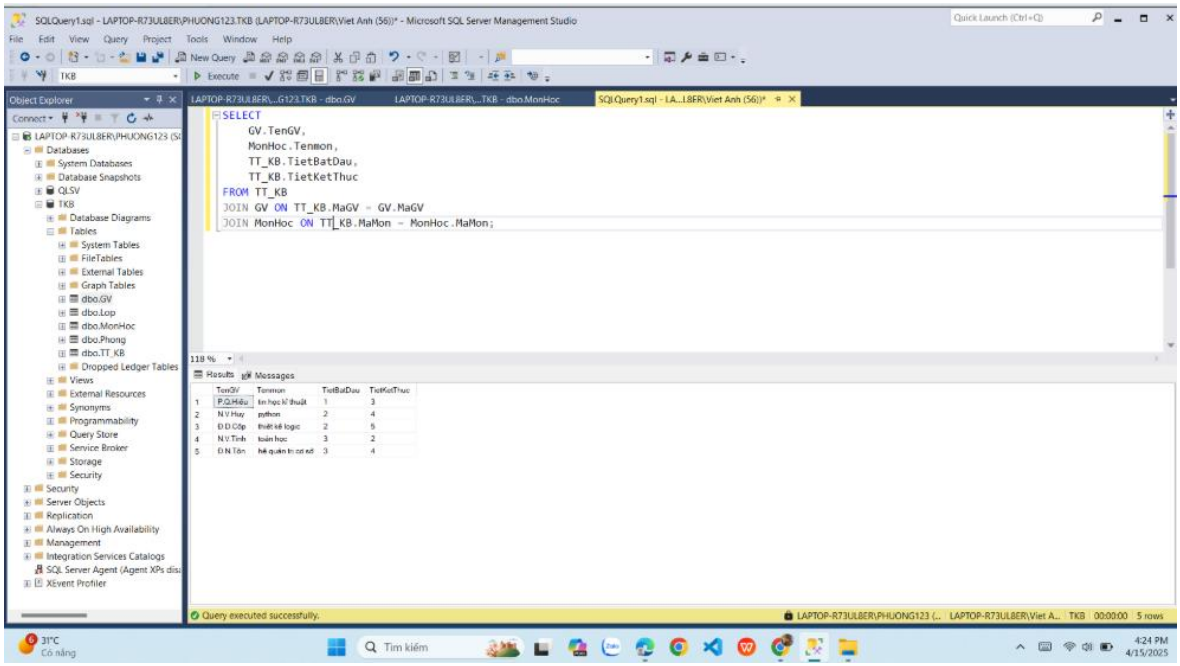


- B2: Thực hiện đặt khóa chính và các khóa ngoại liên kết với khóa chính như trên sơ đồ liên kết:
- B3: Copy dữ liệu từ nguồn :TMS.tnut.edu.vn, paste vào excel sau đó thực hiện dán các dữ liệu cần thiết vào các bảng ở mục edit bảng trong sql

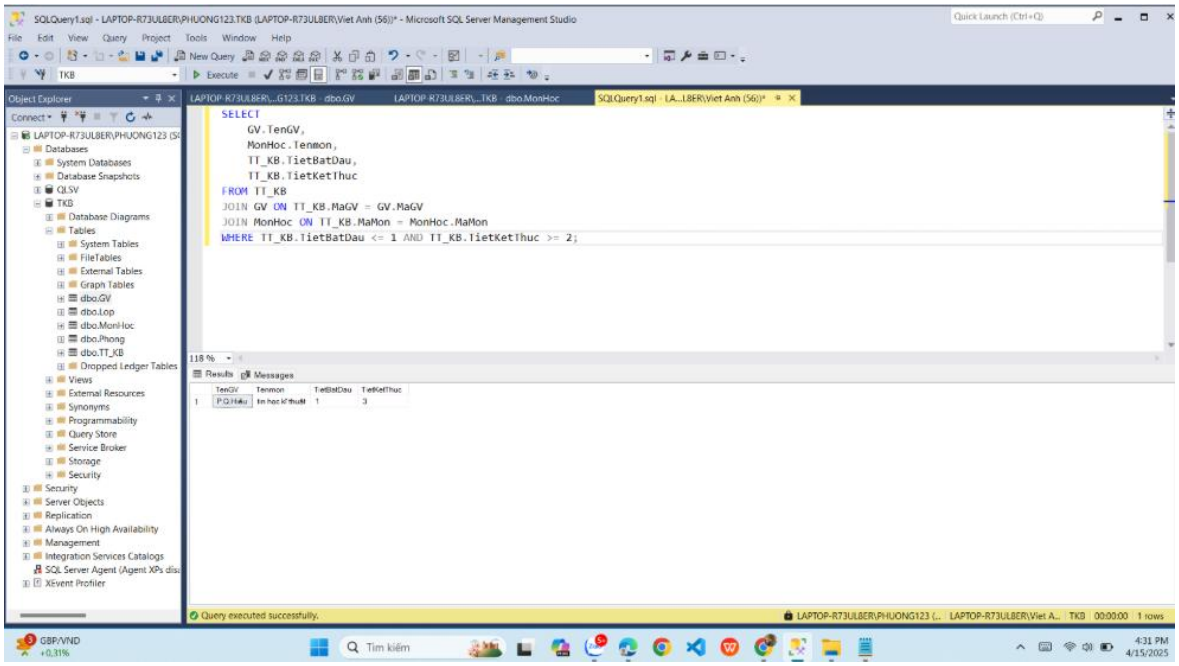




❖ B4: Truy vấn ra thông tin gồm 4 cột: TenGV,TenMon,Tietbandau,Tietketthuc:



❖ trả lời câu hỏi: trong khoảng thời gian từ datetime1 tới datetime2 thì có những gv nào đang bận giảng dạy.



2.1 Code

- Khai báo chương trình
- Sử dụng tham số kiểu date : lọc ngày giảng dạy

```
CREATE OR ALTER PROCEDURE bai_tap_ve_nha_4
   @datetime1 DATETIME,
   @datetime2 DATETIME
```

- Nhập thời gian bắt đầu tiết học và thời gian kết thúc tiết học để đổi số tiết thành giờ cụ thể (giờ vào- giờ ra)

```
WITH TietGio AS (
      SELECT 1 AS tiet, '06:30' AS GioVao, '07:45' AS GioRa UNION ALL
      SELECT 2, '07:55', '09:10' UNION ALL
      SELECT 3, '09:20', '10:35' UNION ALL
      SELECT 4, '10:45', '12:00' UNION ALL
      SELECT 5, '12:30', '13:45' UNION ALL
      SELECT 6, '13:55', '15:10' UNION ALL
      SELECT 7, '15:20', '16:35' UNION ALL
      SELECT 8, '16:45', '18:00' UNION ALL
      SELECT 9, '18:10', '19:25' UNION ALL
      SELECT 10,'19:35', '20:50'
   )
```

- truy vấn các thông tin lịch giảng dạy của các giảng viên trong khoảng thời gian xác định bao gồm
  - tên giảng viên
  - môn dạy
  - giờ vào lớp(tính theo tiết bắt đầu)
  - giờ ra lớp(tính theo tiết kêt thúc)
  - ngày dạy

```
SELECT
      gv.tengv AS HoTenGV,
      mh.tenmon AS MonDay,
      CAST(CONVERT(varchar, tkb.ngay, 23) + ' ' + tg1.GioVao AS
DATETIME) AS GioVao,
      CAST(CONVERT(varchar, tkb.ngay, 23) + ' ' + tg2.GioRa AS
DATETIME) AS GioRa
```

❖ **Select các cột muốn hiện thị**

- from TKB : bảng tkb chứa thông tin
- JOIN gv gv ON tkb.magv = gv.magv : kết nối với bảng gv để lấy thông tin giảng viên
- JOIN monhoc mh ON tkb.mamon = mh.mamon : lấy thông tin môn học
- JOIN TietGio tg1 ON tkb.tietbatdau = tg1.tiet : thời gian bắt đầu tiết học
- JOIN TietGio tg2 ON tkb.tietketthuc = tg2.tiet : Thời gian kết thúc tiết học
- WHERE CAST(CONVERT(varchar, tkb.ngay, 23) + ' ' + tg2.GioRa AS DATETIME) > @datetime1 AND CAST(CONVERT(varchar, tkb.ngay, 23)

+ ' ' + tg1.GioVao AS DATETIME) < @datetime2 tìm các tiết học có thời gian diễn ra giao nhau với khoảng thời gian được truyền vào time1 và time2

```
    FROM
        tkb
        JOIN  gv gv ON tkb.magv = gv.magv
        JOIN monhoc mh ON tkb.mamon = mh.mamon
        JOIN TietGio tg1 ON tkb.tietbatdau = tg1.tiet
        JOIN TietGio tg2 ON tkb.tietketthuc = tg2.tiet
    WHERE
        CAST(CONVERT(varchar, tkb.ngay, 23) + ' ' + tg2.GioRa AS
    DATETIME) > @datetime1 AND
        CAST(CONVERT(varchar, tkb.ngay, 23) + ' ' + tg1.GioVao AS
    DATETIME) < @datetime2
```

- Truyền dữ liệu giảng dạy muốn truy vấn

```
    EXEC bai_tap_ve_nha_4
        @datetime1 = '2025-03-17 06:30:00',
        @datetime2 = '2025-03-18 12:00:00'
```

CODE HOÀN CHỈNH

```
CREATE OR ALTER PROCEDURE bai_tap_ve_nha_4
   @datetime1 DATETIME,
   @datetime2 DATETIME
AS
BEGIN
   SET NOCOUNT ON;

   WITH TietGio AS (
      SELECT 1 AS tiet, '06:30' AS GioVao, '07:45' AS GioRa UNION ALL
      SELECT 2, '07:55', '09:10' UNION ALL
      SELECT 3, '09:20', '10:35' UNION ALL
      SELECT 4, '10:45', '12:00' UNION ALL
      SELECT 5, '12:30', '13:45' UNION ALL
      SELECT 6, '13:55', '15:10' UNION ALL
      SELECT 7, '15:20', '16:35' UNION ALL
      SELECT 8, '16:45', '18:00' UNION ALL
      SELECT 9, '18:10', '19:25' UNION ALL
      SELECT 10,'19:35', '20:50'
   )

   SELECT
      gv.tengv AS HoTenGV,
      mh.tenmon AS MonDay,
      CAST(CONVERT(varchar, tkb.ngay, 23) + ' ' + tg1.GioVao AS DATETIME)
AS GioVao,
      CAST(CONVERT(varchar, tkb.ngay, 23) + ' ' + tg2.GioRa AS DATETIME)
AS GioRa
```

```
    FROM
        tkb
        JOIN  gv gv ON tkb.magv = gv.magv
        JOIN monhoc mh ON tkb.mamon = mh.mamon
        JOIN TietGio tg1 ON tkb.tietbatdau = tg1.tiet
        JOIN TietGio tg2 ON tkb.tietketthuc = tg2.tiet
    WHERE
        CAST(CONVERT(varchar, tkb.ngay, 23) + ' ' + tg2.GioRa AS DATETIME) >
@datetime1 AND
        CAST(CONVERT(varchar, tkb.ngay, 23) + ' ' + tg1.GioVao AS DATETIME)
< @datetime2
END
GO
EXEC bai_tap_ve_nha_4
    @datetime1 = '2025-03-17 06:30:00',
    @datetime2 = '2025-03-18 12:00:00'
```

3. *Mã QR của bài*

Link và QR:
https://github.com/XuanPhuong01/TKB.git

CHƯƠNG 5. BÀI TẬP VỀ NHÀ 5

1. Đề bài

BÀI TẬP VỀ NHÀ 05, Môn Hệ quản trị csdl.
SUBJECT: Trigger on mssql
A. Trình bày lại đầu bài của đồ án PT&TKHT:

1. Mô tả bài toán của đồ án PT&TKHT, đưa ra yêu cầu của bài toán đó
2. Cơ sở dữ liệu của Đồ án PT&TKHT : Có database với các bảng dữ liệu cần thiết (3nf), Các bảng này đã có PK, FK, CK cần thiết

B. Nội dung Bài tập 05:

1. Dựa trên cơ sở là csdl của Đồ án
2. Tìm cách bổ xung thêm 1 (hoặc vài) trường phi chuẩn (là trường tính toán đc, nhưng thêm vào thì ok hơn, ok hơn theo 1 logic nào đó, vd ok hơn về speed) => Nêu rõ logic này!
3. Viết trigger cho 1 bảng nào đó, mà có sử dụng trường phi chuẩn này, nhằm đạt được 1 vài mục tiêu nào đó. => Nêu rõ các mục tiêu
4. Nhập dữ liệu có kiểm soát, nhằm để test sự hiệu quả của việc trigger auto run.
5. Kết luận về Trigger đã giúp gì cho đồ án của em.

HƯỚNG DẪN CÁCH LÀM:
Hướng dẫn làm phần A:

- Chỉ cần nêu ra y/c của đồ án.
- Không cần chụp quá trình làm ra db, tables.
- Chỉ cần đưa ra db gồm các bảng nào, mỗi bảng có các trường nào, kiểu dữ liệu nào, và pk, fk, ck của các bảng.

Hướng dẫn làm phần B:

1. Sv tạo repo mới trên github, cho phép truy cập public.
2. Tạo file Readme.md, đầu file để thông tin cá nhân sv.
3. Tiếp theo đưa phần A vào file Reame.md .
4. Các thao tác làm trên csdl bằng phần mềm ssms.
5. Chụp ảnh màn hình quá trình làm.
6. Paste ngay vào Readme.md, rồi gõ mô tả ảnh này làm gì, nhập gì, hay đạt được điều gì...
7. Có thể thêm những nhận xét hoặc kết luận cho việc bản thân đã hiểu rõ thêm về 1 vấn đề gì đó.
8. Lặp lại các step 4 5 6 cho đến khi hoàn thành yêu cầu của phần B.
9. Xuất các file sql chứa cấu trúc và data, up lên cùng repo.
10. Link đến repo cần mở được trực tiếp nội dung, Paste link này vào file excel online ghim trên nhóm. Thầy sẽ dùng tool để check các link này.

DEADLINE: 23H59:59 NGÀY 23/04/2025
p/s:

- Sv được phép tham khảo mọi nguồn, nhưng phải tự làm lại.
- Đọc thêm nội quy học tập để biết các chế tài.
- Đã đến lúc khẳng định bản thân và toả sáng!
- Chỗ nào vướng mắc cứ share lên nhóm để cùng tháo gỡ.

3. Bài làm

- Yêu cầu đồ án: Phân tích thiết kế hệ thống hỗ trợ tân sinh viên chọn nghành học tại trường ĐHKTCN
- Dựa trên yêu cầu của hệ thống hỗ trợ tân sinh viên chọn nghành học tại trường ĐHKTCN, em có các bảng dữ liệu cần thiết(3NF) như sau:

- Bảng CauHoi:

LAPTOP-R73UL8ER\...gành - dbo.CauHoi

Column Name	Data Type	Allow Nulls
MaCH	varchar(10)	☐
NoiDung	nvarchar(200)	☑
LinhVuc	nvarchar(50)	☑
		☐

- Bảng GoiYNganh:

LAPTOP-R73UL8ER\...h - dbo.GoiYNganh

Column Name	Data Type	Allow Nulls
MaSV	varchar(10)	☑
MaNganh	varchar(10)	☑
MucDoPhuHop	float	☑
		☐

- Bảng Nganh:

LAPTOP-R73UL8ER\...gành - dbo.Nganh

Column Name	Data Type	Allow Nulls
MaNganh	varchar(10)	☐
TenNganh	nvarchar(20)	☑
KhoiXetTuyen	varchar(10)	☑
MoTa	nvarchar(300)	☑
		☐

- Bảng SinhVien:

LAPTOP-R73UL8ER\P...nh - dbo.SinhVien

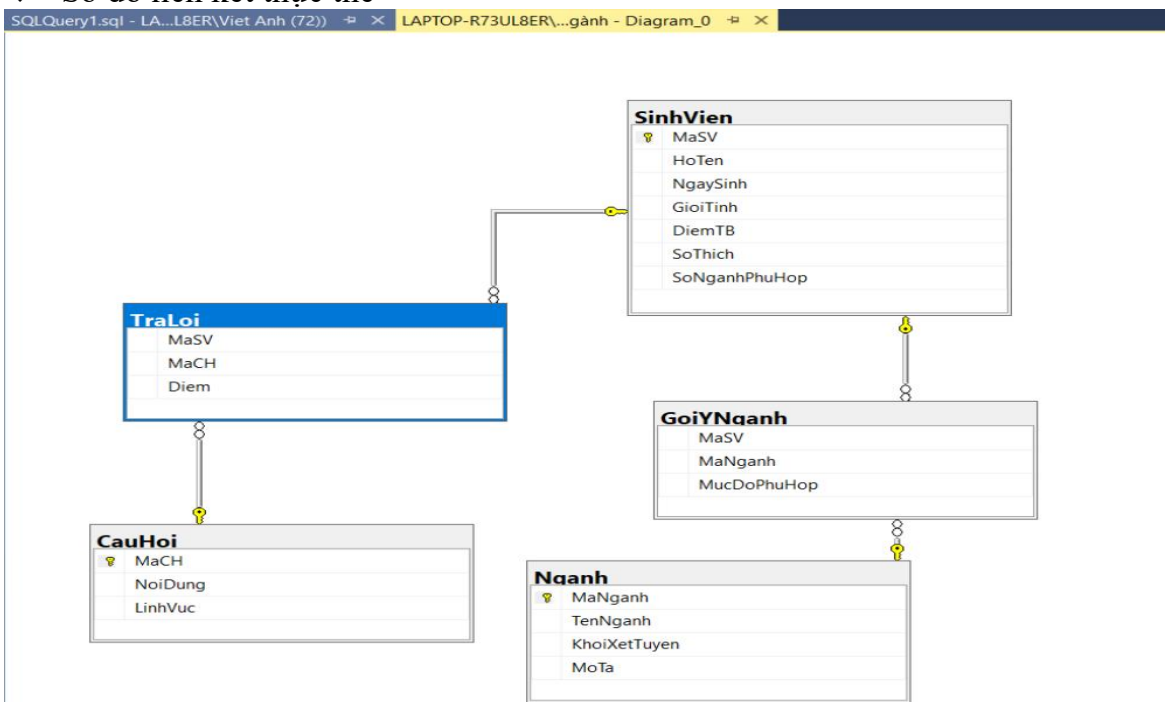
Column Name	Data Type	Allow Nulls
MaSV	varchar(10)	☐
HoTen	nvarchar(20)	☑
NgaySinh	date	☑
GioiTinh	nvarchar(10)	☑
DiemTB	float	☑
SoThich	nvarchar(100)	☑
SoNganhPhuHop	int	☑
		☐

❖ Bang TraLoi:

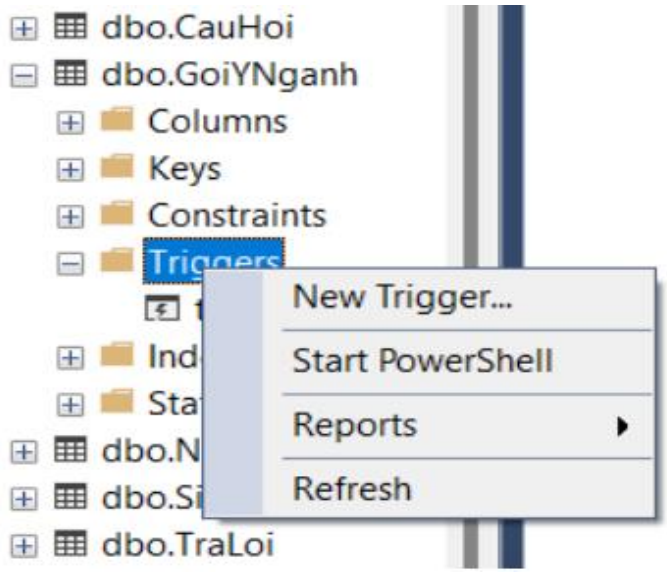
LAPTOP-R73UL8ER\P...gành - dbo.TraLoi

Column Name	Data Type	Allow Nulls
MaSV	varchar(10)	☑
MaCH	varchar(10)	☑
Diem	float	☑
		☐

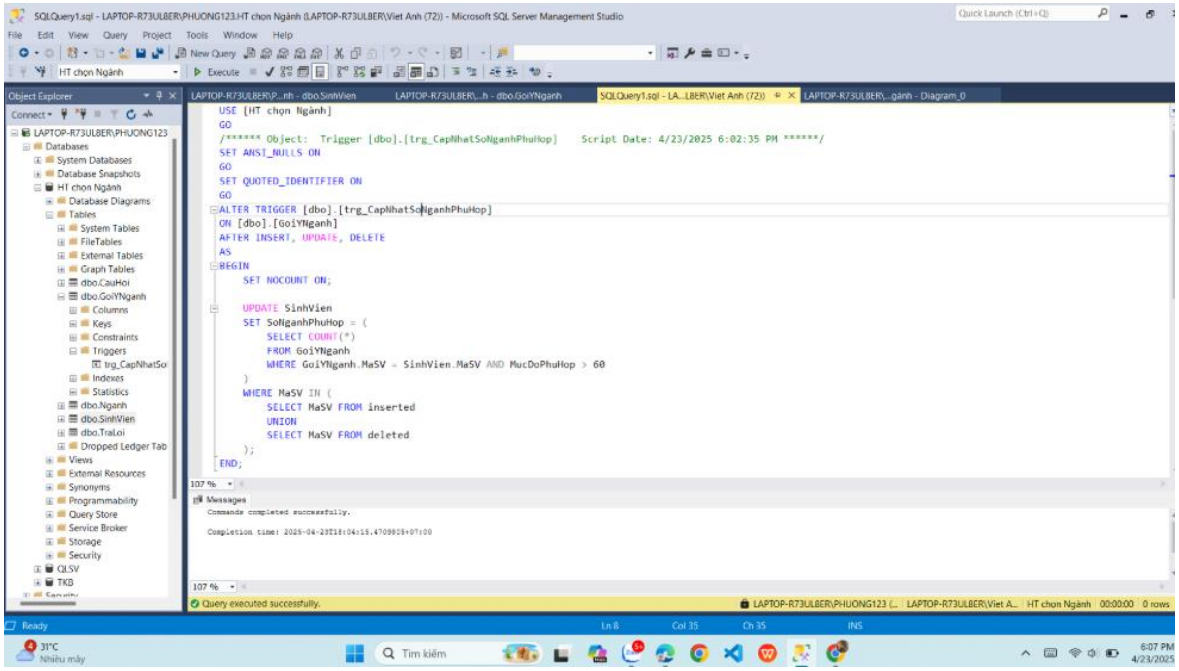
❖ Sơ đồ liên kết thực thể



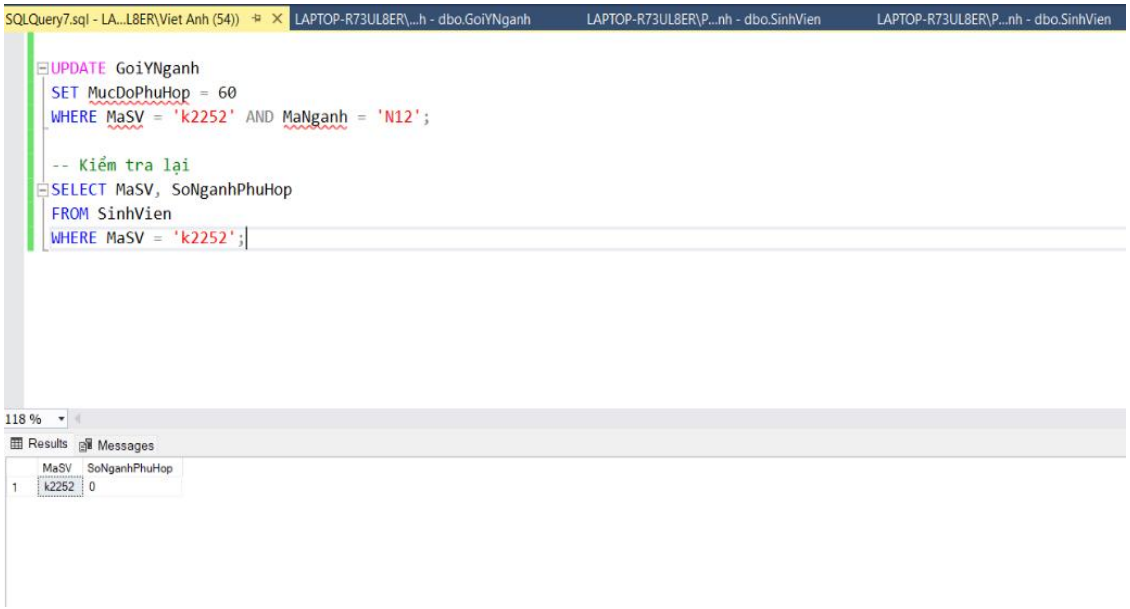
❖ Bấm dấu "+" vào bảng và chuột phải vào Triggers ---> new trigger :



❖ Tự động cập nhật cột SoNganhPhuHop trong bảng SinhVien mỗi khi có thêm, sửa, hoặc xóa dữ liệu ở bảng GoiYNganh:



❖ Test:



❖ Code hoàn chỉnh:

```
CREATE TABLE [dbo].[CauHoi](
        [MaCH] [varchar](10) NOT NULL,
        [NoiDung] [nvarchar](200) NULL,
        [LinhVuc] [nvarchar](50) NULL,
CONSTRAINT [PK_CauHoi] PRIMARY KEY CLUSTERED
(
        [MaCH] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

USE [HT chọn Ngành]
GO

/****** Object:  Table [dbo].[GoiYNganh]    Script Date: 4/23/2025 6:47:46 PM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO
```

```
CREATE TABLE [dbo].[GoiYNganh](
	[MaSV] [varchar](10) NULL,
	[MaNganh] [varchar](10) NULL,
	[MucDoPhuHop] [float] NULL
) ON [PRIMARY]
GO

ALTER TABLE [dbo].[GoiYNganh] WITH CHECK ADD CONSTRAINT [FK_GoiYNganh_Nganh] FOREIGN KEY([MaNganh])
REFERENCES [dbo].[Nganh] ([MaNganh])
GO

ALTER TABLE [dbo].[GoiYNganh] CHECK CONSTRAINT [FK_GoiYNganh_Nganh]
GO

ALTER TABLE [dbo].[GoiYNganh] WITH CHECK ADD CONSTRAINT [FK_GoiYNganh_SinhVien] FOREIGN KEY([MaSV])
REFERENCES [dbo].[SinhVien] ([MaSV])
GO

ALTER TABLE [dbo].[GoiYNganh] CHECK CONSTRAINT [FK_GoiYNganh_SinhVien]
GO

ALTER TABLE [dbo].[GoiYNganh] WITH CHECK ADD CONSTRAINT [CK_GoiYNganh] CHECK (([MucDoPhuHop]>=(0) AND [MucDoPhuHop]<=(100)))
GO

ALTER TABLE [dbo].[GoiYNganh] CHECK CONSTRAINT [CK_GoiYNganh]
GO

USE [HT chọn Ngành]
GO

/****** Object: Table [dbo].[Nganh] Script Date: 4/23/2025 6:49:24 PM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Nganh](
	[MaNganh] [varchar](10) NOT NULL,
	[TenNganh] [nvarchar](20) NULL,
```

```
	[KhoiXetTuyen] [varchar](10) NULL,
	[MoTa] [nvarchar](300) NULL,
 CONSTRAINT [PK_Nganh] PRIMARY KEY CLUSTERED
(
	[MaNganh] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

USE [HT chọn Ngành]
GO

/****** Object:  Table [dbo].[SinhVien]    Script Date: 4/23/2025 6:49:41 PM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[SinhVien](
	[MaSV] [varchar](10) NOT NULL,
	[HoTen] [nvarchar](20) NULL,
	[NgaySinh] [date] NULL,
	[GioiTinh] [nvarchar](10) NULL,
	[DiemTB] [float] NULL,
	[SoThich] [nvarchar](100) NULL,
	[SoNganhPhuHop] [int] NULL,
 CONSTRAINT [PK_SinhVien] PRIMARY KEY CLUSTERED
(
	[MaSV] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SinhVien] WITH CHECK ADD CONSTRAINT [CK_SinhVien] CHECK (([ngaysinh]<=getdate()))
GO

ALTER TABLE [dbo].[SinhVien] CHECK CONSTRAINT [CK_SinhVien]
GO
```

```
USE [HT chọn Ngành]
GO

/****** Object:  Table [dbo].[TraLoi]    Script Date: 4/23/2025 6:50:09 PM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[TraLoi](
	[MaSV] [varchar](10) NULL,
	[MaCH] [varchar](10) NULL,
	[Diem] [float] NULL
) ON [PRIMARY]
GO

ALTER TABLE [dbo].[TraLoi] WITH CHECK ADD CONSTRAINT [FK_TraLoi_CauHoi] FOREIGN KEY([MaCH])
REFERENCES [dbo].[CauHoi] ([MaCH])
GO

ALTER TABLE [dbo].[TraLoi] CHECK CONSTRAINT [FK_TraLoi_CauHoi]
GO

ALTER TABLE [dbo].[TraLoi] WITH CHECK ADD CONSTRAINT [FK_TraLoi_SinhVien] FOREIGN KEY([MaSV])
REFERENCES [dbo].[SinhVien] ([MaSV])
GO

ALTER TABLE [dbo].[TraLoi] CHECK CONSTRAINT [FK_TraLoi_SinhVien]
GO


USE [HT chọn Ngành]
GO
/****** Object:  Trigger [dbo].[trg_CapNhatSoNganhPhuHop]    Script Date: 4/23/2025 6:50:28 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER TRIGGER [dbo].[trg_CapNhatSoNganhPhuHop]
ON [dbo].[GoiYNganh]
AFTER INSERT, UPDATE, DELETE
```

```
AS
BEGIN
   SET NOCOUNT ON;

   UPDATE SinhVien
   SET SoNganhPhuHop = (
      SELECT COUNT(*)
      FROM GoiYNganh
      WHERE GoiYNganh.MaSV = SinhVien.MaSV AND MucDoPhuHop > 60
   )
   WHERE MaSV IN (
      SELECT MaSV FROM inserted
      UNION
      SELECT MaSV FROM deleted
   );
END;
```

2. *Mã QR của bài*

❖ Link và mã QR:
https://github.com/XuanPhuong01/BAI-TAP-5.git

CHƯƠNG 6: BÀI TẬP VỀ NHÀ 6

1. Đề bài

Bài tập 6: Hệ quản trị CSDL Chủ đề: Câu lệnh Select Yêu cầu bài tập: Cho file sv_tnut.sql (1.6MB)

1. Hãy nêu các bước để import được dữ liệu trong sv_tnut.sql vào sql server của em
2. dữ liệu đầu vào là tên của sv; sđt; ngày, tháng, năm sinh của sinh viên (của sv đang làm bài tập này)
3. nhập sql để tìm xem có những sv nào trùng hoàn toàn ngày/tháng/năm với em?
4. nhập sql để tìm xem có những sv nào trùng ngày và tháng sinh với em?
5. nhập sql để tìm xem có những sv nào trùng tháng và năm sinh với em?
6. nhập sql để tìm xem có những sv nào trùng tên với em?
7. nhập sql để tìm xem có những sv nào trùng họ và tên đệm với em.
8. nhập sql để tìm xem có những sv nào có sđt sai khác chỉ 1 số so với sđt của em.
9. BẢNG SV CÓ HƠN 9000 ROWS, HÃY LIỆT KÊ TẤT CẢ CÁC SV NGÀNH KMT, SẮP XẾP THEO TÊN VÀ HỌ ĐỆM, KIỂU TIẾNG VIỆT, GIẢI THÍCH.
10. HÃY NHẬP SQL ĐỂ LIỆT KÊ CÁC SV NỮ NGÀNH KMT CÓ TRONG BẢNG SV (TRÌNH BÀY QUÁ TRÌNH SUY NGHĨ VÀ GIẢI NHỮNG VỨNG MẮC)

DEADLINE: 23H59:59 NGÀY 25/4/2025

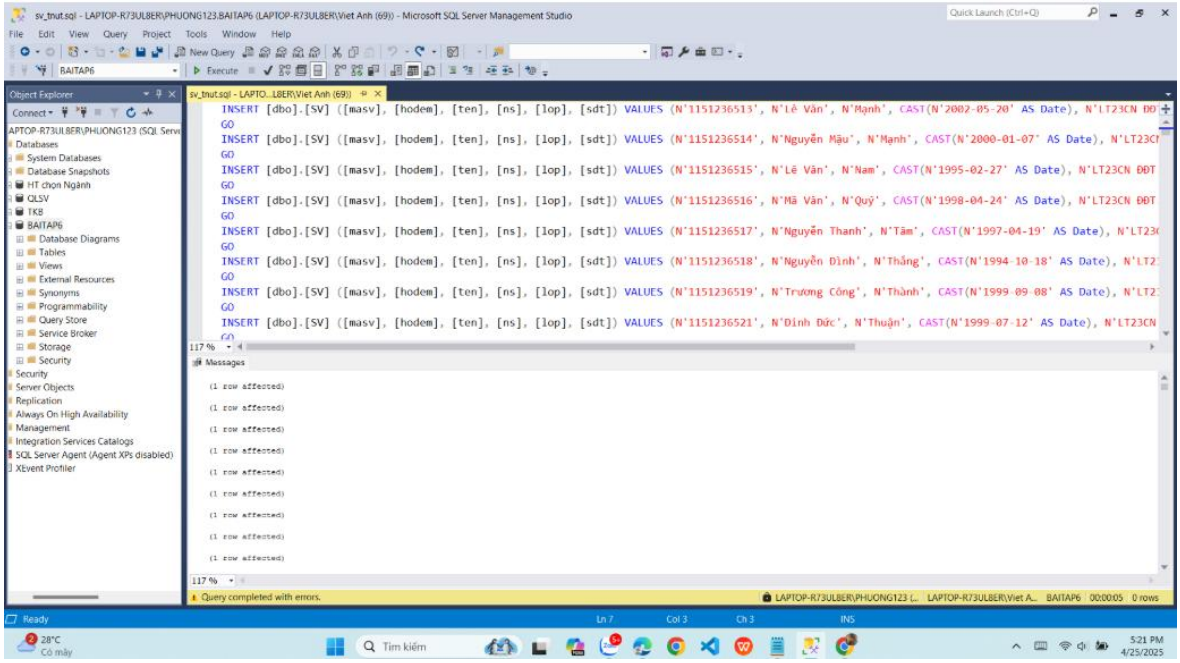
Ghi chú: Giải thích tại sao lại có SQL như vậy.

2. Bài làm

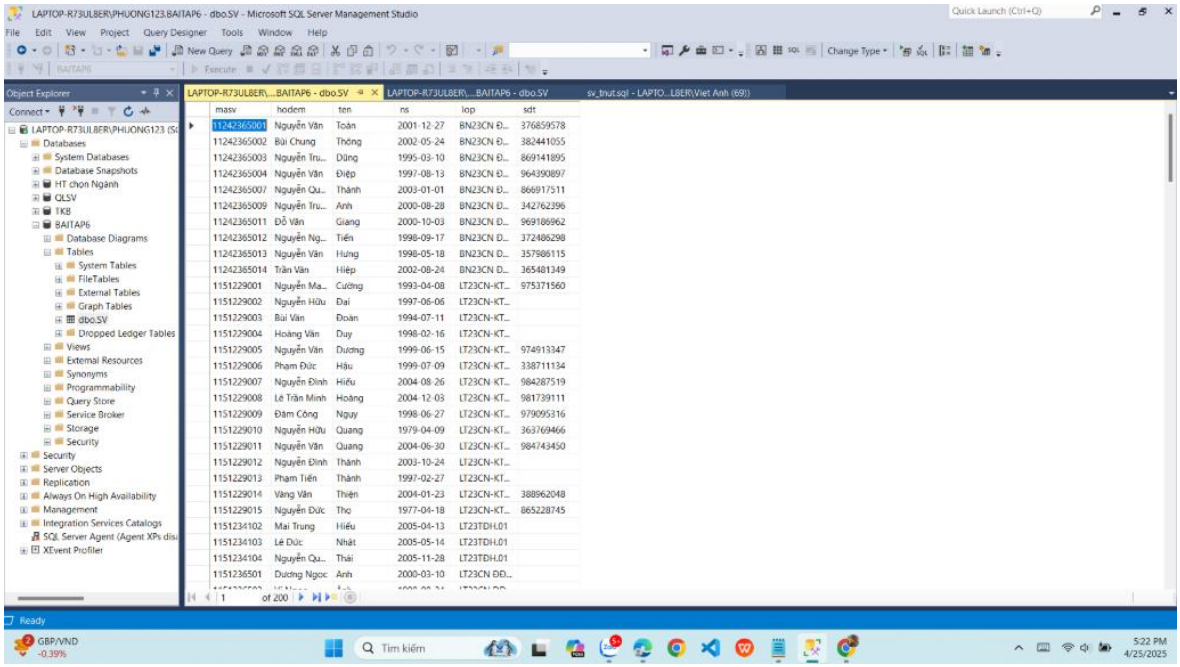
❖ Câu 1:

- Bước 1: Mở phần mền sql
- Bước 2 Tạo database mới, sau khi tạo xong chọn database sv_tnut ở góc trái tên ssms để đảm bảo thực hiện script đúng nới
- Bước3: vào menu: file -> open-> file. chọn đến file sv_tnut.sql. Toàn bộ cript sẽ hiện lên màn hình
- Bước 4: chạy toàn bộ cậu lệnh SQL
- Bước 5. sau khi chạy xong sẽ thấy dữ liệu sinh viên đã được import vào bẳng SV

- để import được dữ liệu trong sv_tnut.sql thì đầu tiên em tải file sv_tnut.sql về máy. Sau đó tạo 1 database mới mang tên sv_tnut rồi em mở file sv_tnut.sql vừa tải về -> execute toàn bộ file sv_tnut.sql:



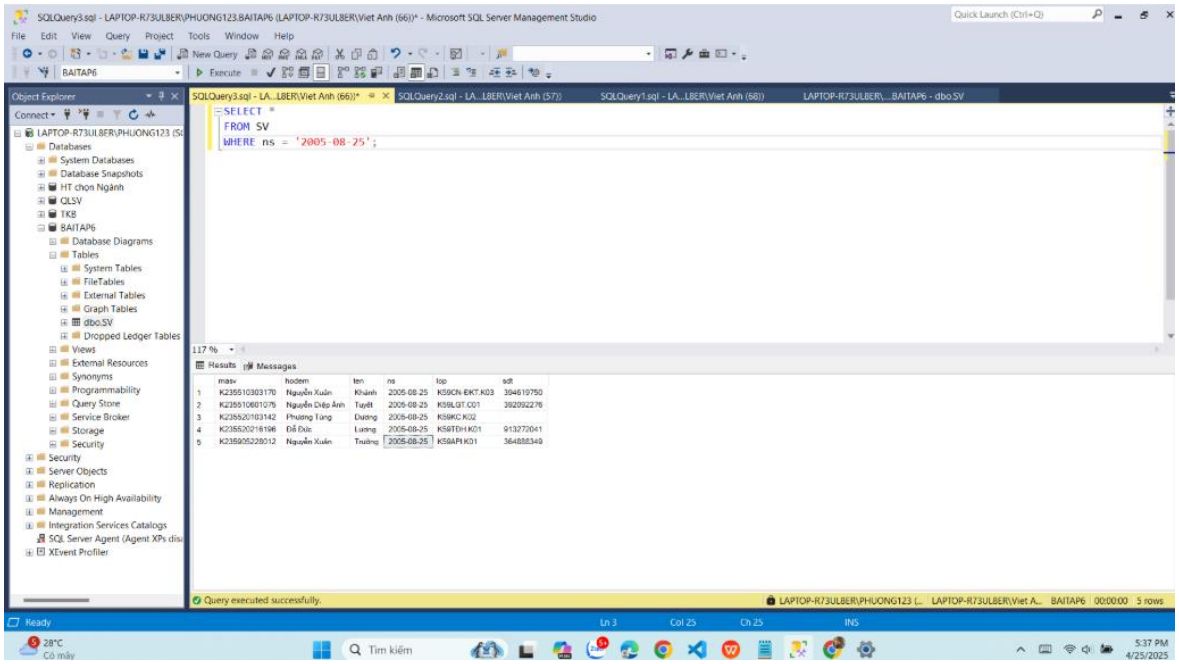
- Sau khi đã chạy em được bảng dữ liệu dbo.SV như hình:



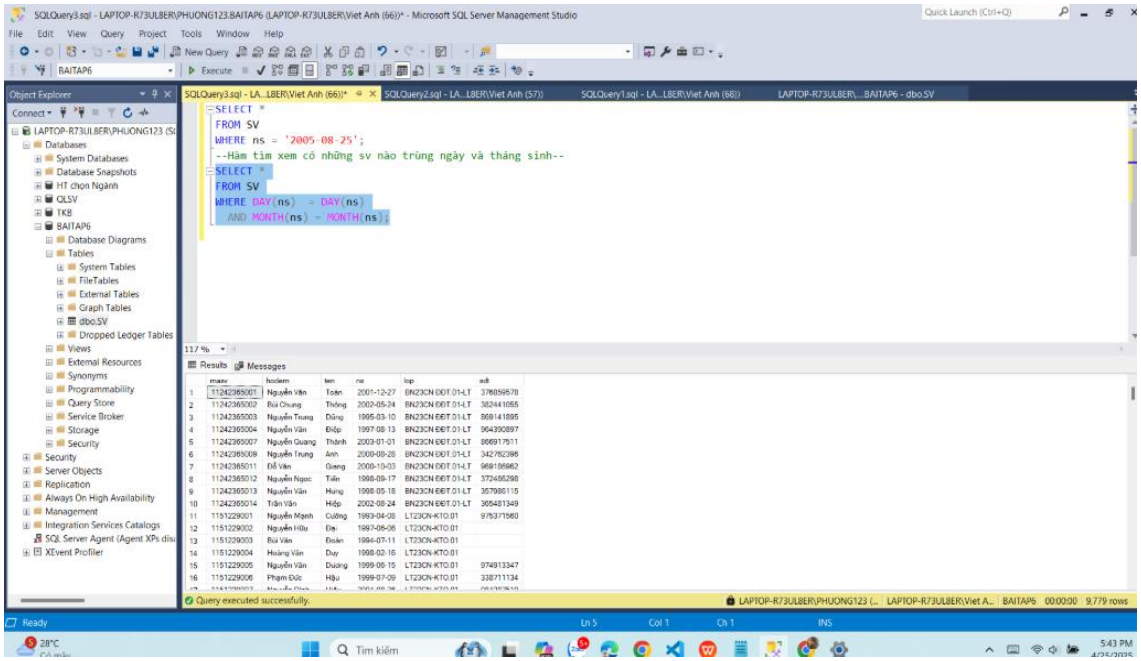
- Câu 2: dữ liệu đầu vào là tên của sv; sđt; ngày, tháng, năm sinh của sinh viên (của sv đang làm bài tập này)
- Code

```
SELECT * FROM SV
WHERE CONVERT(DATE, Ns, 103) = '2004-8-25';
SELECT * FROM SV
WHERE DAY(NS) = 25 AND MONTH(NS) = 8;
SELECT * FROM SV
WHERE MONTH(NS) = 8 AND YEAR(NS) = 2004;
SELECT * FROM SV
WHERE Ten = N'Phương';
SELECT * FROM SV
WHERE HoDem = N' Nguyễn Thị Xuân ';
```

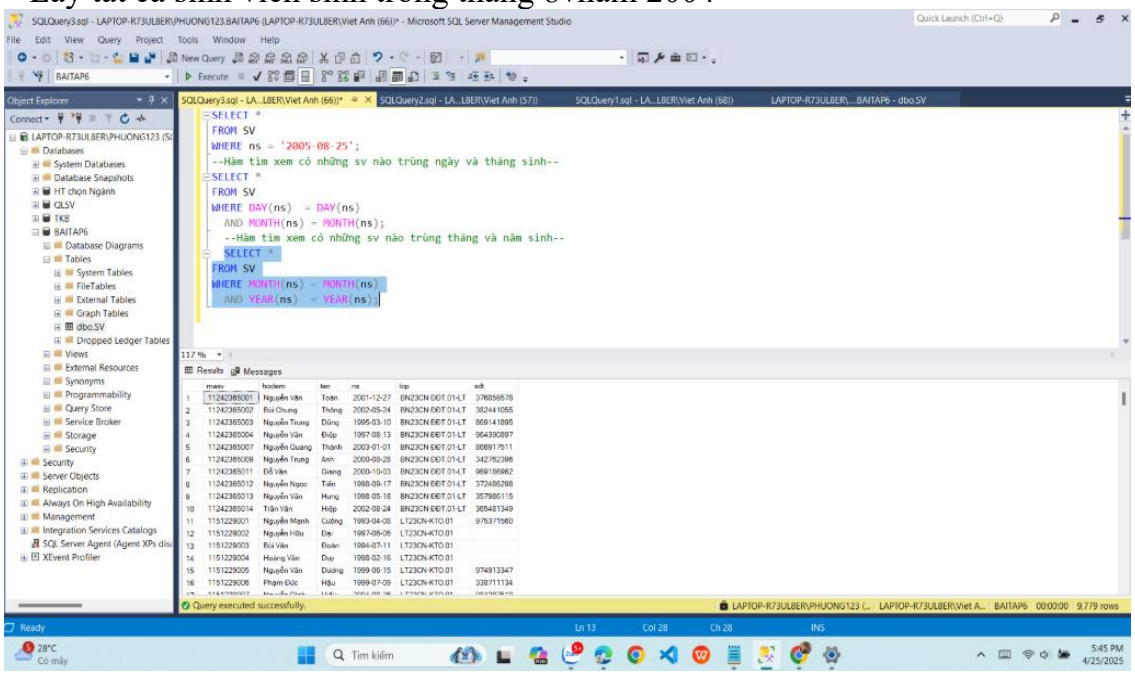
- Câu 3. nhập sql để tìm xem có những sv nào trùng hoàn toàn ngày/tháng/năm với em?
  - Lọc sinh viên có ngày sinh 25/08/2004
  - ns là cột ngày tháng năm sinh của các sinh viên -CONVERT(DATE, Ns, 103) = '2004-08-25'; chuyển ngày theo định dạng dd/mm/yyyy (kiểu 103) sang kiểu date để so sánh
  - Trả về sinh viên có ngày sinh là 25/08/2004



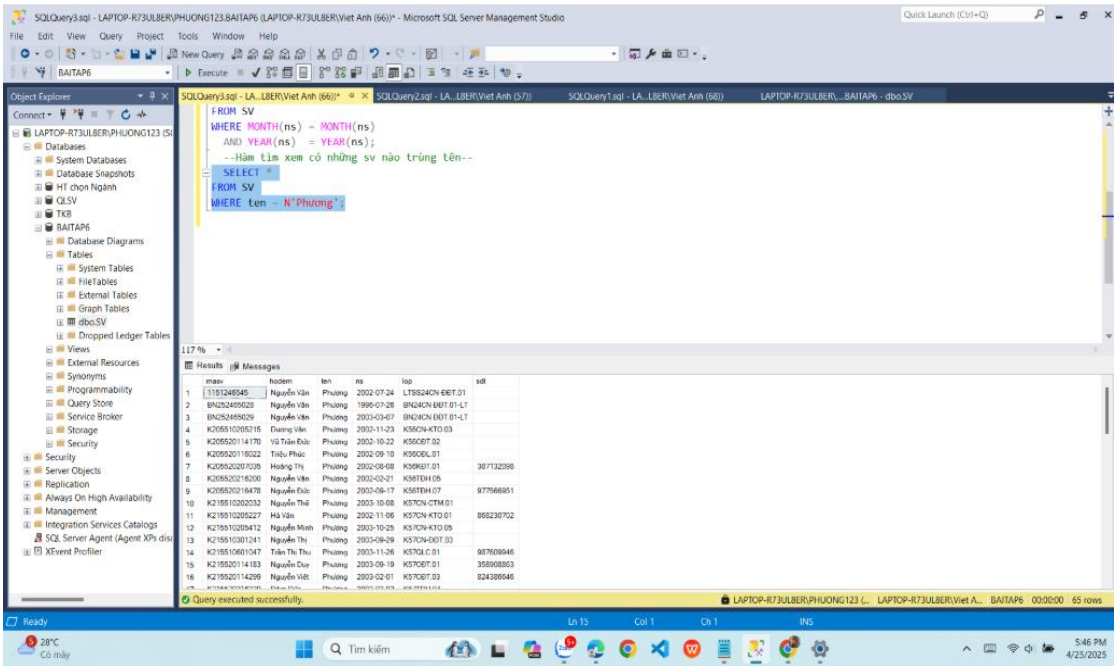
❖ Câu 4. nhập sql để tìm xem có những sv nào trùng ngày và tháng sinh với em? dùng hàm Day() và Month để so sánh ngày và tháng riêng lẻ



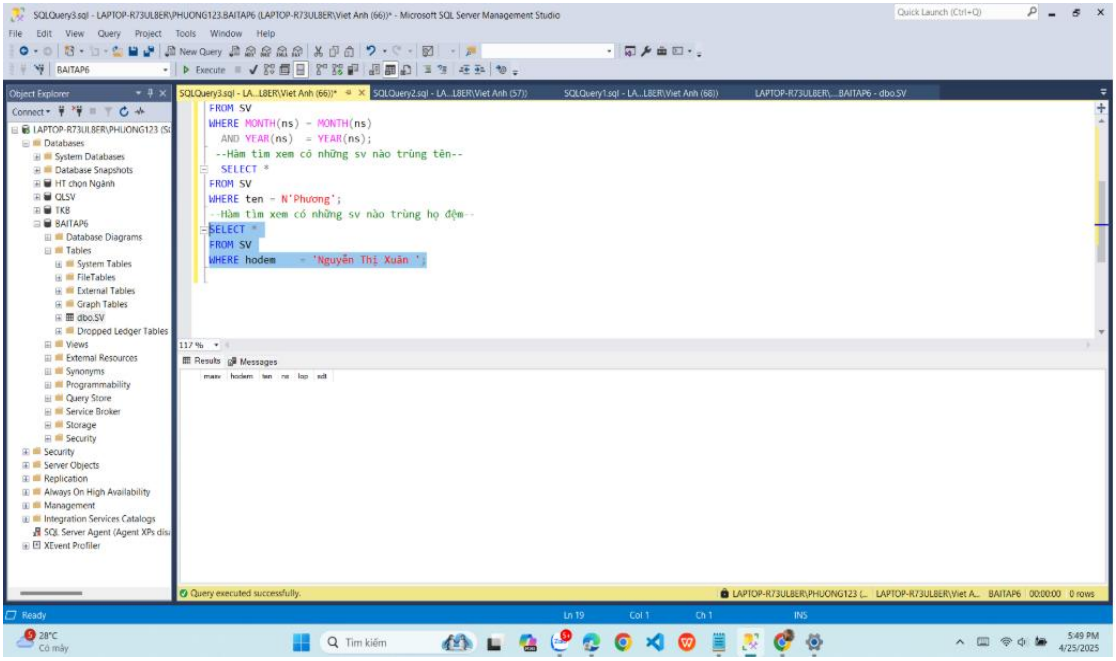
❖ Câu 5. nhập sql để tìm xem có những sv nào trùng tháng và năm sinh với em? Lấy tất cả sinh viên sinh trong tháng 8vnăm 2004



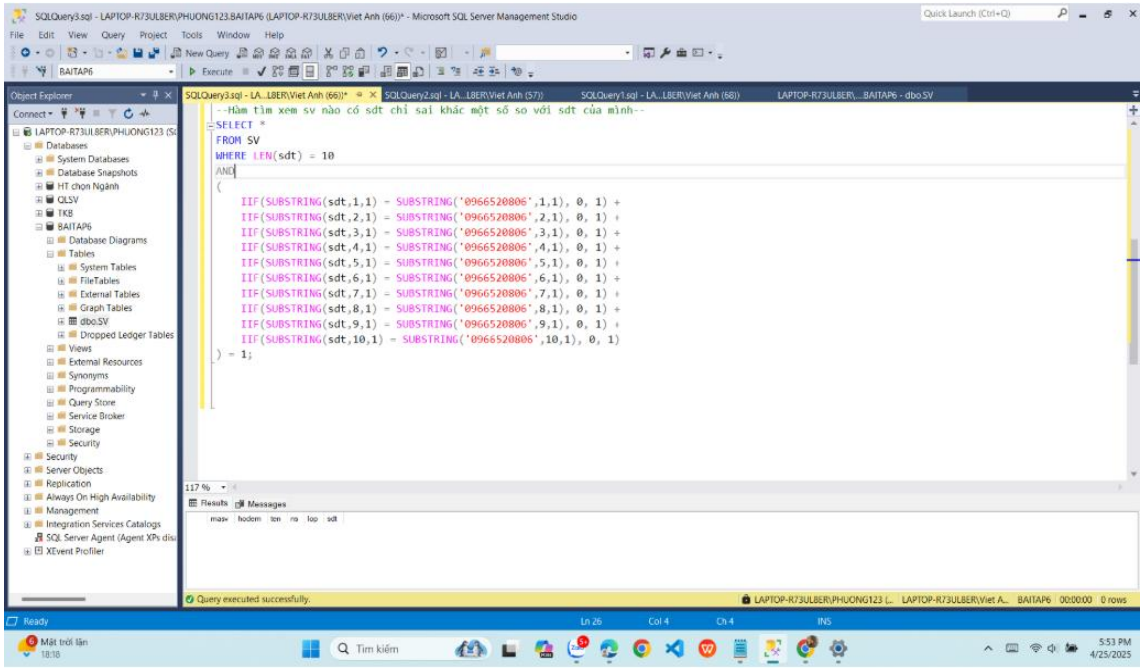
❖ Câu 6. nhập sql để tìm xem có những sv nào trùng tên với em?
- Lọc sinh viên có tên riêng là Trang
- Dùng N'' để hôc trợ unicode (chữ có dấu)



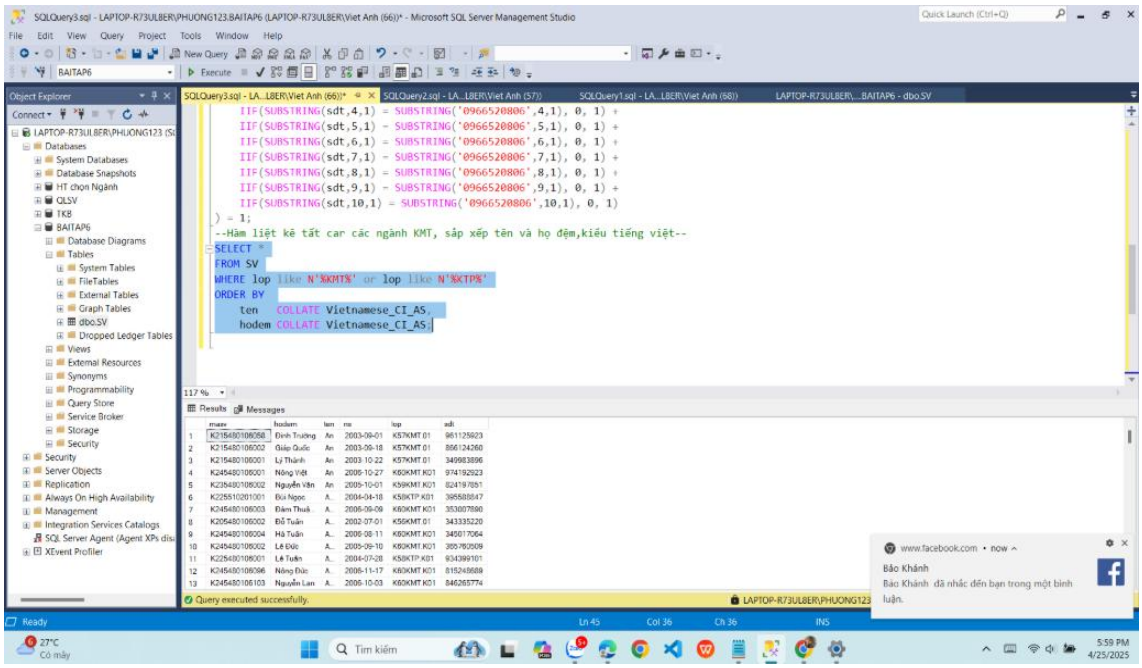
❖ Câu 7. nhập sql để tìm xem có những sv nào trùng họ và tên đệm với em. Lọc theo họ và tên đệm 'Nguyễn Thị Xuân'



❖ Câu 8: nhập sql để tìm xem có những sv nào có sđt sai khác chỉ 1 số so với sđt của em.
Kiểm tra độ dài số điện thoại là 9 số

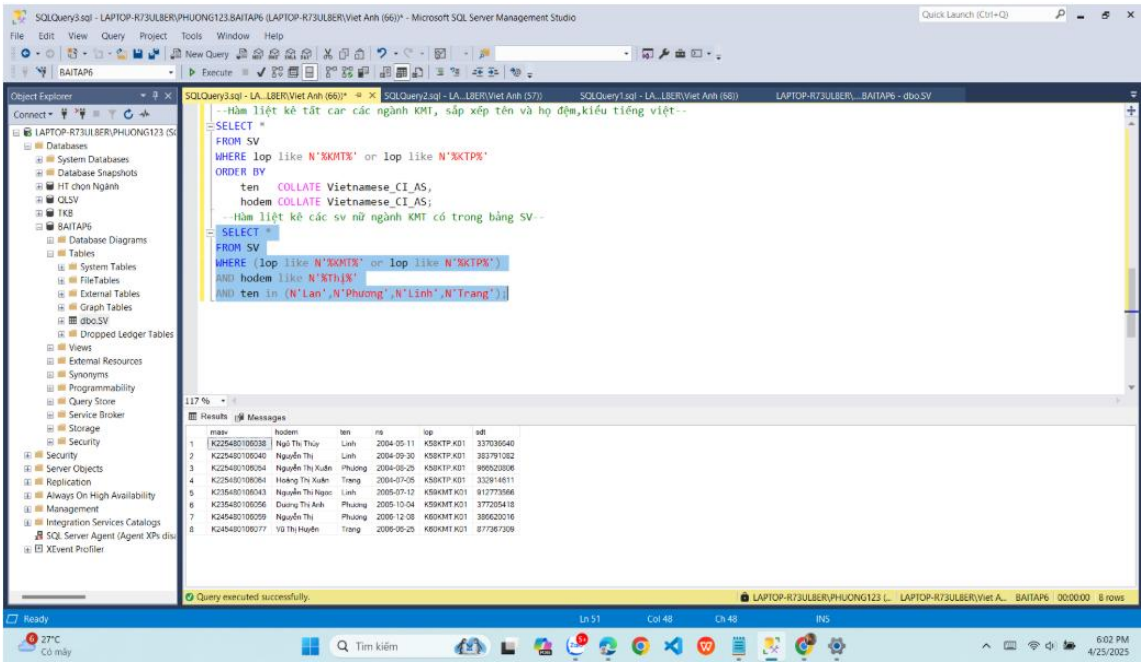


❖ Câu 9. BẢNG SV CÓ HƠN 9000 ROWS, HÃY LIỆT KÊ TẤT CẢ CÁC SV NGÀNH KMT, SẮP XẾP THEO TÊN VÀ HỌ ĐỆM, KIỂU TIẾNG VIỆT, GIẢI THÍCH.
Lọc sinh viên lớp có tên chứa KMT hoặc KTP
vì trong file sv_tnut không có trường giới tính nên ta thêm điều kiện : họ đệm có chữ "thị" hoặc có 1 tên trong các tên nữ phổ biến ở Việt Nam



❖ Câu 10. HÃY NHẬP SQL ĐỂ LIỆT KÊ CÁC SV NỮ NGÀNH KMT CÓ TRONG BẢNG SV (TRÌNH BÀY QUÁ TRÌNH SUY NGHĨ VÀ GIẢI NHỮNG VỨNG MẮC)

- Lọc sinh viên lớp chứa KMT
- sắp xếp theo tên và họ đệm
- COLLATE Vietnamese_CI_AI: sắp xếp không phân biệt dấu và hoa thường
- CI: Case insensitive (không phân biệt hoa/ thường)
- AI: Accent Insensitive (không phân biệt dấu)



CODE HOÀN CHỈNH

```
SELECT * FROM SV
WHERE CONVERT(DATE, Ns, 103) = '2004-08-25';
SELECT * FROM SV
WHERE DAY(NS) = 25 AND MONTH(NS) = 08;
SELECT * FROM SV
WHERE MONTH(NS) = 08 AND YEAR(NS) = 2004;
SELECT * FROM SV
WHERE Ten = N'Phương';
SELECT * FROM SV
WHERE HoDem = N'Nguyễn Thị Xuân';
SELECT * FROM SV
WHERE LEN(SDT) = 9
AND (
   IIF(SUBSTRING(SDT,1,1) <> '0', 1, 0) +
   IIF(SUBSTRING(SDT,2,1) <> '3', 1, 0) +
   IIF(SUBSTRING(SDT,3,1) <> '3', 1, 0) +
   IIF(SUBSTRING(SDT,4,1) <> '2', 1, 0) +
   IIF(SUBSTRING(SDT,5,1) <> '9', 1, 0) +
   IIF(SUBSTRING(SDT,6,1) <> '1', 1, 0) +
```

```
    IIF(SUBSTRING(SDT,7,1) <> '4', 1, 0) +
    IIF(SUBSTRING(SDT,8,1) <> '6', 1, 0) +
    IIF(SUBSTRING(SDT,9,1) <> '1', 1, 0) +
    IIF(SUBSTRING(SDT,10,1) <> '1', 1, 0)
) = 1;

SELECT *FROM SV
WHERE
    (lop LIKE '%KMT%' OR lop LIKE '%KTP%')
    AND (
        hodem LIKE N'%Thị%'
        OR ten IN (
            N'My', N'Trinh', N'Tuyết', N'Thảo',
            N'Tiên', N'Như', N'Thu', N'Yến', N'Vân', N'Hạnh'
        )
    );
          SELECT *
FROM SV
WHERE lop LIKE '%KMT%'
ORDER BY ten COLLATE Vietnamese_CI_AI, hodem COLLATE
Vietnamese_CI_AI;
```

3. *Mã QR của bài*

❖ Link và mã QR:

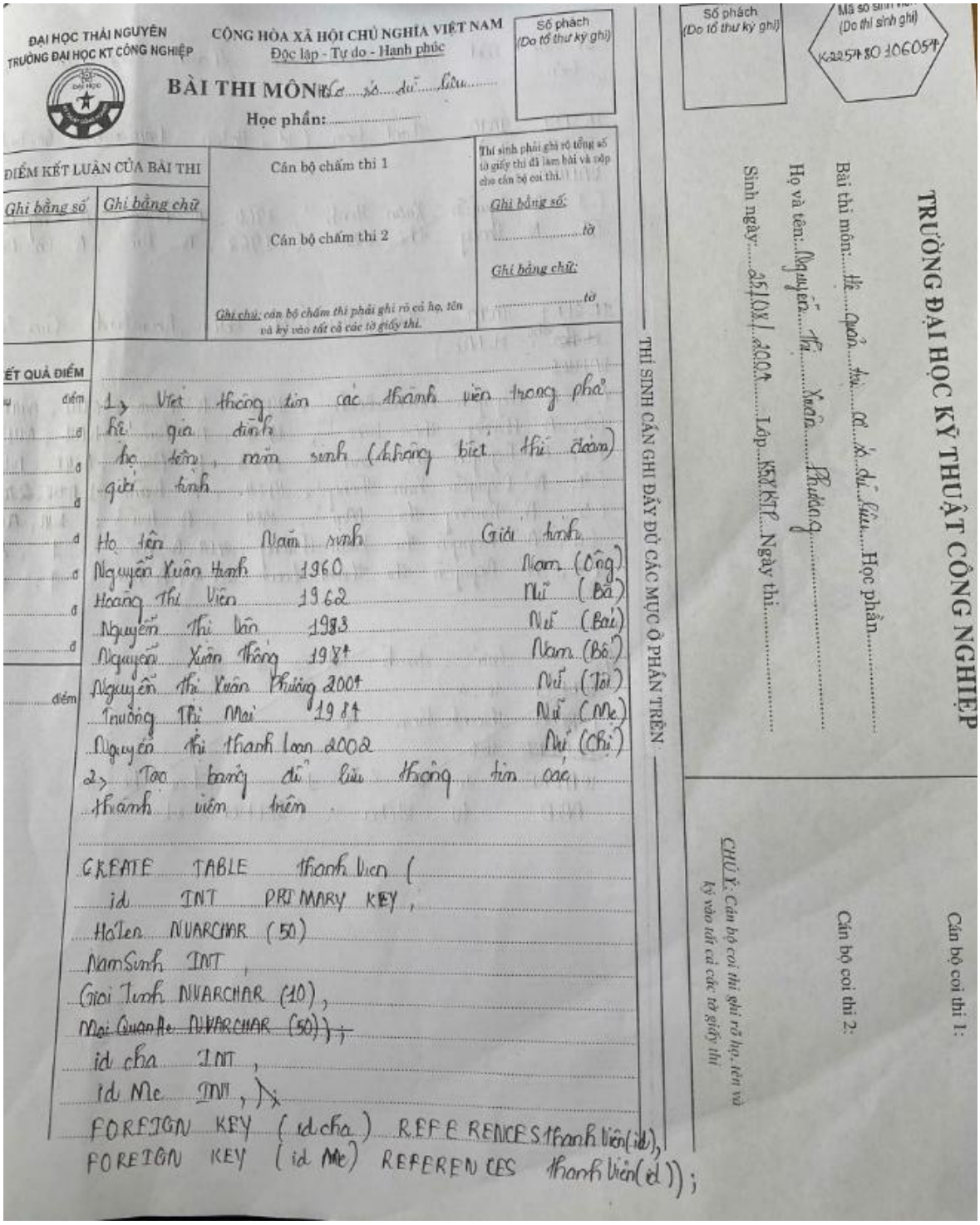
https://github.com/XuanPhuong01/BTVN-SO-6.git

CHƯƠNG 7: BÀI KIỂM TRA

1. Đề bài

- Viết thông tin các thành viên trong phả hệ gia đình ho. tên, năm sinh,, giới tính, bố mẹ của người này
- Tạo bảng để lưu thông tin thành viên
- Tạo lệnh insert các thành viên vào bảng trên
- Truy vấn xem ai là người mà ta gọi là chú, bác, ông nội, bà nội, hoặc vài thành viên khác input 1 là id của ta output 1 là thông tin các thành viên là chú input 2 là id của ta output 2 là thông tin các thành viên là Bác .....

2. Bài làm



ĐẠI HỌC THÁI NGUYÊN
TRƯỜNG ĐẠI HỌC KT CÔNG NGHIỆP

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

Số phách (Do tổ thư ký ghi)

BÀI THI MÔN: Hệ cơ sở dữ liệu
Học phần:

ĐIỂM KẾT LUẬN CỦA BÀI THI		Cán bộ chấm thi 1	Thí sinh phải ghi rõ tổng số tờ giấy thi đã làm bài và nộp cho cán bộ coi thi.
Ghi bằng số	Ghi bằng chữ	Cán bộ chấm thi 2	Ghi bằng số: ..... tờ
		Ghi chú: cán bộ chấm thi phải ghi rõ cả họ, tên và ký vào tất cả các tờ giấy thi.	Ghi bằng chữ: ..... tờ

KẾT QUẢ ĐIỂM

điểm

1> Viết thông tin các thành viên trong phả hệ gia đình
họ tên, năm sinh (không biết thì đoán)
giới tính

Họ tên	Năm sinh	Giới tính
Nguyễn Xuân Hanh	1960	Nam (Ông)
Hoàng Thị Viên	1962	Nữ (Bà)
Nguyễn Thị Vân	1983	Nữ (Bái)
Nguyễn Xuân Thắng	1984	Nam (Bố)
Nguyễn Thị Xuân Phương	2004	Nữ (Tôi)
Trương Thị Mai	1984	Nữ (Mẹ)
Nguyễn Thị Thanh Loan	2002	Nữ (Chị)

2> Tạo bảng dữ liệu thông tin các thành viên trên

```
CREATE TABLE thanh_vien (
    id INT PRIMARY KEY,
    HoTen NVARCHAR (50)
    NamSinh INT,
    Gioi_Tinh NVARCHAR (10),
    ~~Mai_QuanHe NVARCHAR (50),~~
    id_cha INT,
    id_Me INT,
    FOREIGN KEY (id_cha) REFERENCES thanh_vien(id),
    FOREIGN KEY (id_Me) REFERENCES thanh_vien(id));
```

THÍ SINH CẤN GHI ĐẦY ĐỦ CÁC MỤC Ở PHẦN TRÊN

Số phách (Do tổ thư ký ghi)

Mã số sinh viên (Do thí sinh ghi)
K225480106054

TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP

Bài thi môn: Hệ quản trị cơ sở dữ liệu Học phần:
Họ và tên: Nguyễn Thị Xuân Phương
Sinh ngày: 25/10/2004 Lớp: K58KTP Ngày thi:

Cán bộ coi thi 1:
Cán bộ coi thi 2:
CHÚ Ý: Cán bộ coi thi ghi rõ họ, tên và ký vào tất cả các tờ giấy thi

3, Tạo lệnh để insert các thành viên vào bảng trên

```
INSERT INTO Thanh_Vien (id, HoTen, NamSinh, GioiTinh,
Mối_Quan_He)
VALUES
(1, N'Nguyễn Xuân Hình', 1960, N'Nam', N'Ông Nội'),
(2, N'Hoàng Thị Viên', 1962, N'Nữ', N'Bà Nội')
```

```
INSERT INTO Thanh_Vien (id, HoTen, NamSinh, GioiTinh,
idBố, idMẹ)
VALUES
(1, N'Nguyễn Xuân Hình', 1960, N'Nam', NULL, NULL), -- Ông Nội
(2, N'Hoàng Thị Viên', 1962, N'Nữ', NULL, NULL), -- Bà Nội
(3, N'Nguyễn Thị Vân', 1983, N'Nữ', NULL, NULL), -- Bác gái
(4, N'Nguyễn Xuân Thắng', 1984, N'Nam', 1, 2), -- Bố
(5, N'Trương Thị Mai', 1984, N'Nữ', NULL, NULL), -- Mẹ
(6, N'Nguyễn Thị Xuân Phương', 2004, N'Nữ', 4, 5), -- Tôi
(7, N'Nguyễn Thị Thanh Loan', 2002, N'Nữ', 4, 5); -- Chị gái
```

4, -- Giả định id = 3 là tôi

```
SELECT tv.*
FROM Thanh_Vien tv
WHERE tv.id <> 1 -- Khác bố tôi
AND tv.idCha = (SELECT id
```

3. Mã QR của bài

❖ Link và mã QR:
https://github.com/XuanPhuong01/bai-kiem-tra-mon-he-QTCSDL.git

KẾT LUẬN

Qua quá trình thực hiện các bài tập và tiểu luận môn Hệ quản trị cơ sở dữ liệu, em đã tích lũy được nhiều kiến thức và kỹ năng thực tiễn quan trọng. Em đã hiểu rõ hơn về quy trình xây dựng cơ sở dữ liệu từ thiết kế sơ bộ đến cài đặt thực tế, sử dụng ngôn ngữ SQL để tạo bảng, thao tác dữ liệu, viết trigger, thủ tục và xử lý truy vấn phức tạp. Đồng thời, việc triển khai các bài tập trên hệ thống SQL Server và quản lý phiên bản trên GitHub đã giúp em nâng cao khả năng làm việc chuyên nghiệp và tự học trong môi trường công nghệ.

Em xin gửi lời cảm ơn chân thành đến thầy giáo ThS. Đỗ Duy Cốp đã tận tình hướng dẫn và hỗ trợ em trong suốt quá trình học tập. Em cũng xin cảm ơn các thầy cô trong bộ môn và bạn bè đã tạo điều kiện để em hoàn thành tốt tiểu luận này. Dù vẫn còn một số hạn chế, em tin rằng kết quả đạt được là nền tảng vững chắc cho những bước phát triển sau này trong lĩnh vực công nghệ thông tin.