TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP

KHOA ĐIỆN TỬ

BÀI TẬP LỚN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

NGÀNH : KỸ THUẬT PHẦN MỀM

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

HỌ VÀ TÊN SINH VIÊN	:	Nguyễn Thị Hằng Nga
MSSV	:	K225480106050
LỚP	:	K58KTP

THÁI NGUYÊN - 2025

NHẬN XÉT CỦA GIÁO VIÊN

Sinh viên 1: Nguyễn Thị Hằng Nga

Lớp **:** K58KTP

NHẬN XÉT CỦA GIÁO VIÊN

……

……

……

…….....

……

…….....

Xếp loại:…… Điểm :……

Thái Nguyên, ngày….tháng…..năm 20....

CHỮ KÝ CỦA GIÁO VIÊN

(Ký và ghi rõ họ tên)

MỤC LỤC

MỤC LỤC 2
LỜI CẢM ƠN 3
LỜI CAM ĐOAN 4
CHƯƠNG 1 : CÀI ĐẶT SQL SEVER 2022 5
1.1 YÊU CẦU 5
1.2 CÁC BƯỚC THỰC HIỆN 6
1.3 LINK TRA CỨU 8
CHƯƠNG 2 TẠO CƠ SỞ DỮ LIỆU VÀ BẢNG 9
2.1 YÊU CẦU 9
2.2 CÁC BƯỚC THỰC HIỆN 10
2.3 LINK TRA CỨU 22
CHƯƠNG 3 TRUY VẤN DỮ LIỆU 23
3.1 YÊU CẦU 23
2.2 CÁC BƯỚC THỰC HIỆN 24
3.3 LINK TRA CỨU 37
CHƯƠNG 4 TẠO QUERY TRUY VẤN 38
4.1 YÊU CẦU 38
4.2 CÁC BƯỚC THỰC HIỆN 38
4.3 LINK TRA CỨU 43
CHƯƠNG 5 TRINGGER 44
5.1 YÊU CẦU 44
5.2 CÁC BƯỚC THỰC HIỆN 44
5.3 LINK TRA CỨU 57
CHƯƠNG 6 TRUY VẤN DỮ LIỆU 58
6.1 YÊU CẦU 58
6.2 CÁC BƯỚC THỰC HIỆN 58
6.3 LINK TRA CỨU 64

LỜI CẢM ƠN

Tên em là Nguyễn Thị Hằng Nga, học viên trong lớp 58KTP môn Hệ quản trị cơ sở dữ liệu. Nhân dịp kết thúc khóa học, em xin gửi lời cảm ơn chân thành đến thầy vì đã tận tâm giảng dạy và chia sẻ kiến thức quý báu trong suốt thời gian qua.

Nhờ sự hướng dẫn của thầy, em đã hiểu rõ hơn về các khái niệm như mô hình dữ liệu, ngôn ngữ truy vấn SQL, thiết kế cơ sở dữ liệu và quản lý giao dịch. Những bài giảng dễ hiểu của thầy đã giúp em nắm vững lý thuyết và áp dụng vào thực tế một cách hiệu quả.

Không chỉ truyền đạt kiến thức chuyên môn, thầy còn là tấm gương về sự nhiệt huyết và trách nhiệm trong nghề. Em học được từ thầy không chỉ kiến thức mà còn cả thái độ làm việc nghiêm túc và tinh thần cầu tiến.

Một lần nữa, em xin cảm ơn thầy vì tất cả những gì thầy đã dành cho lớp học. Em hy vọng sẽ có cơ hội được học hỏi thêm từ thầy trong tương lai.

Chúc thầy luôn mạnh khỏe, hạnh phúc và thành công trong công tác giảng dạy.

Trân trọng !

LỜI CAM ĐOAN

Tôi xin cam đoan bài tập lớn này là công trình nghiên cứu của chúng em. Các số liệu sử dụng trong luận văn là trung thực. Các kết quả nghiên cứu được trình bày trong đồ án chưa từng được công bố tại bất kỳ công trình nào khác.

Tên sinh viên

Nguyễn Thị Hằng Nga

CHƯƠNG 1 : CÀI ĐẶT SQL SEVER 2022

1.1 YÊU CẦU

1. Sinh viên mô tả các bước trong quá trình cài đặt SQL Server 2022 Dev: (download link nào, đc file gì, khi chạy thì chọn vào chỗ nào để download file iso bản dev, có iso rồi thì làm gì để chạy đc setup.exe trong nó, trong giao diện của setup thì chọn vào đâu, ý nghĩa mỗi tuỳ chọn này, ... => kết quả được sql server đang chạy dạng service: kiểm tra service này đang running ở đâu?

2. Các bước download và cài đặt phần mềm sql managerment studio (bản mới nhất)

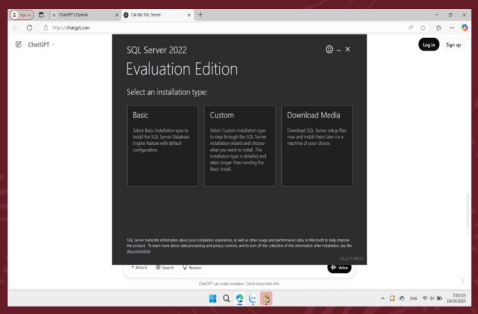
3. Các bước để đăng nhập từ sql managerment studio vào Sql server đã cài (thử cả 2 cách: Windows auth và Sql server Auth)

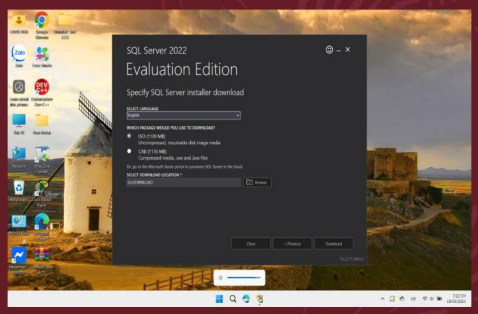
4. (option, nhưng muốn đạt 10 điểm thì làm thêm bước này)

- Cấu hình dynamic port và mở port trên firewall: để cho phép truy cập trong mạng LAN từ máy tính khác vào sql server (dễ, ko cần gợi ý)
- Cấu hình mở port trên router để có thể truy cập qua internet từ 1 máy tính ở bất kỳ đâu tới Sql server đang cài trên máy tính cá nhân (gợi ý: NAT Port, Port Forward)
- Trường hợp ko có quyền thao tác nat port với router (như trong ktx) thì cần làm gì để có thể truy cập vào sql server từ xa (gợi ý keyword: VPN tunnel)

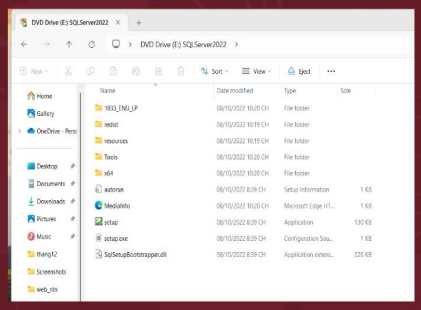
1.2 CÁC BƯỚC THỰC HIỆN

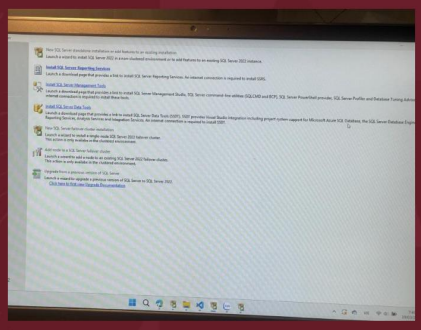
Bước 2: Tải xuống file ISO cài đặt Chạy file SQL2022-SSEI-Dev.exe.
Ở giao diện SQL Server Installation Center chọn Download Media. Chọn định dạng ISO → Nhấn Download
. Chờ tải xong, bạn sẽ có file SQLServer2022-x64-Dev.iso.

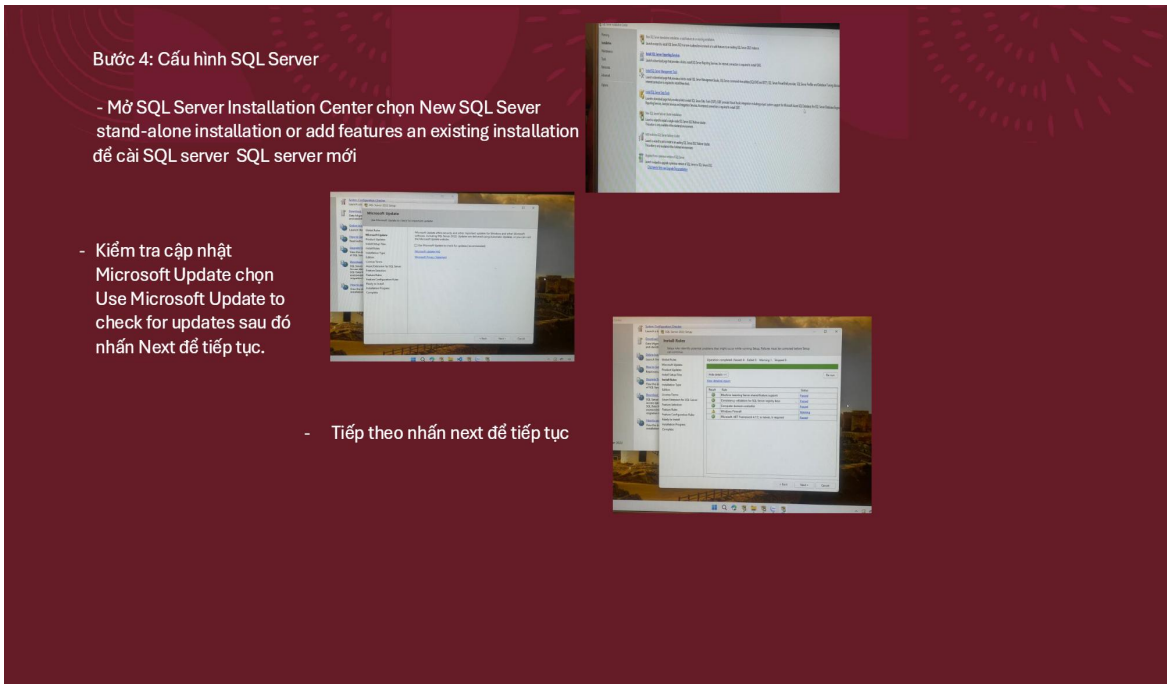




Bước 3: Chạy bộ cài từ file ISO Mount file ISO bằng cách click đúp vào file hoặc nhấn chuột phải → Mount. Vào ổ đĩa ảo mới xuất hiện (ví dụ: E:\) → Chạy setup.exe. SQL Server Installation Center hiện ra → Chọn New SQL Server stand-alone installation.







1.3 LINK TRA CỨU

1 Link github : [Nganguyen211/baitap1]

2 Mã qr

CHƯƠNG 2 TẠO CƠ SỞ DỮ LIỆU VÀ BẢNG

2.1 YÊU CẦU

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)

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.

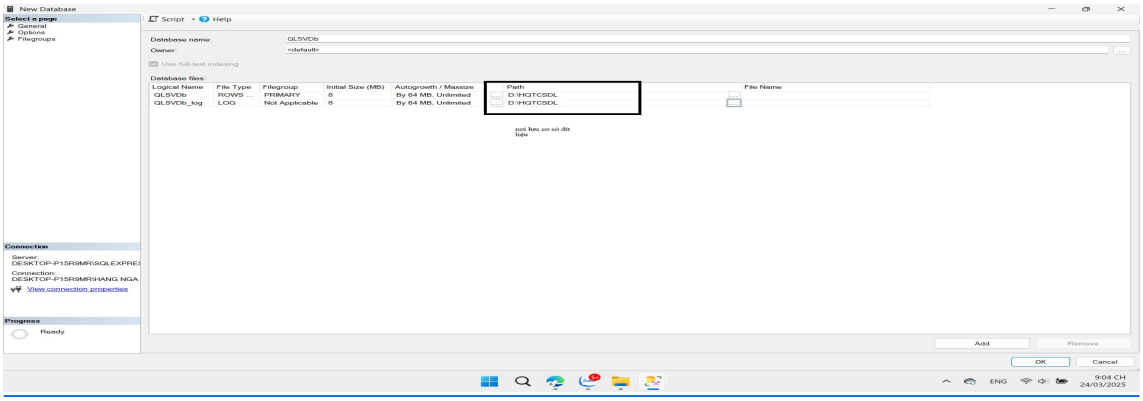
  + Tạo các bảng dữ liệu với các trường như mô tả, chọn kiểu dữ liệu phù hợp với thực tế (tự tìm hiểu)

  + 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

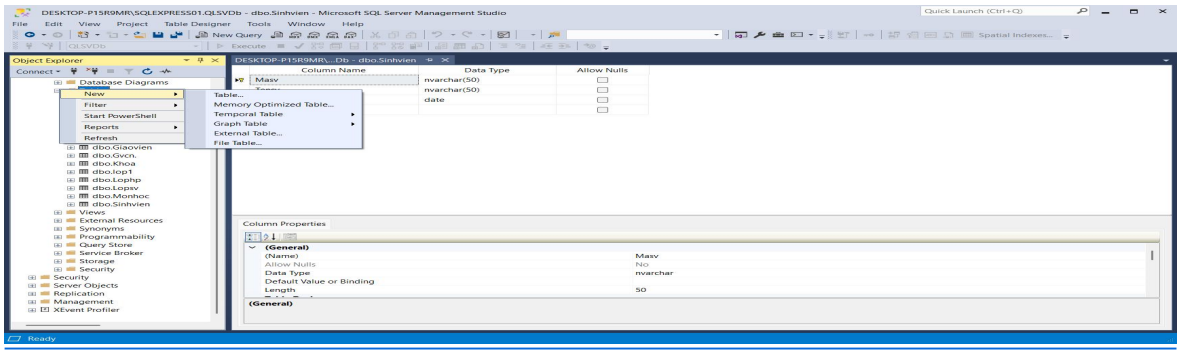
2.2 CÁC BƯỚC THỰC HIỆN

Bước đầu tạo cơ sở dữ liệu quản lý sinh viên

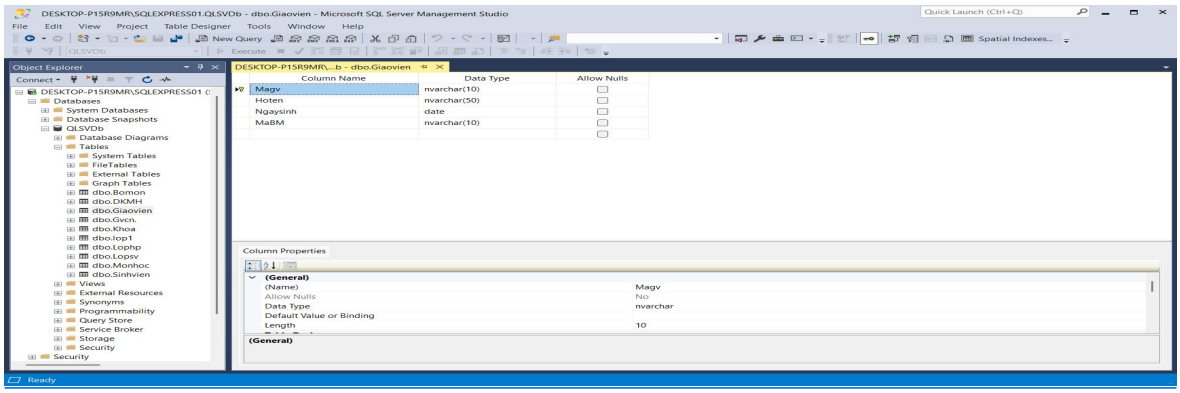


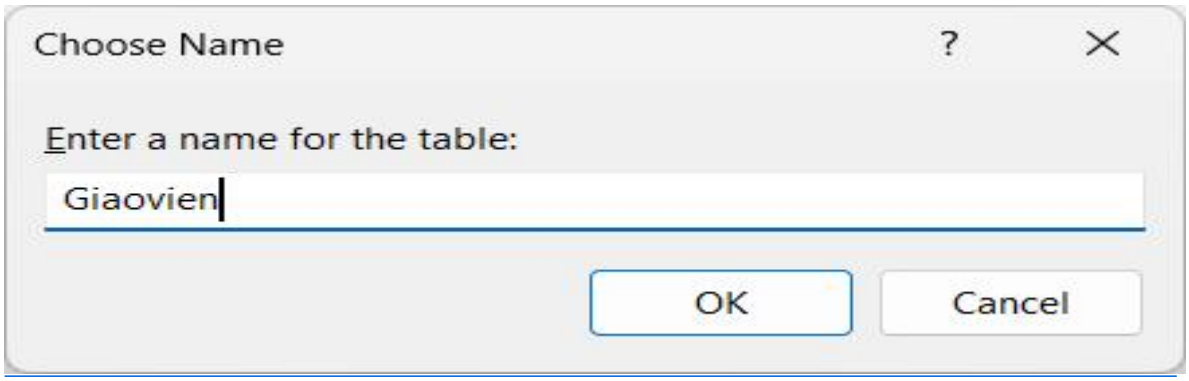
Trong đ ó:Database name (tên csdl_QLSV) và default là quyền sở hữu csdl thuộc về người tạo database

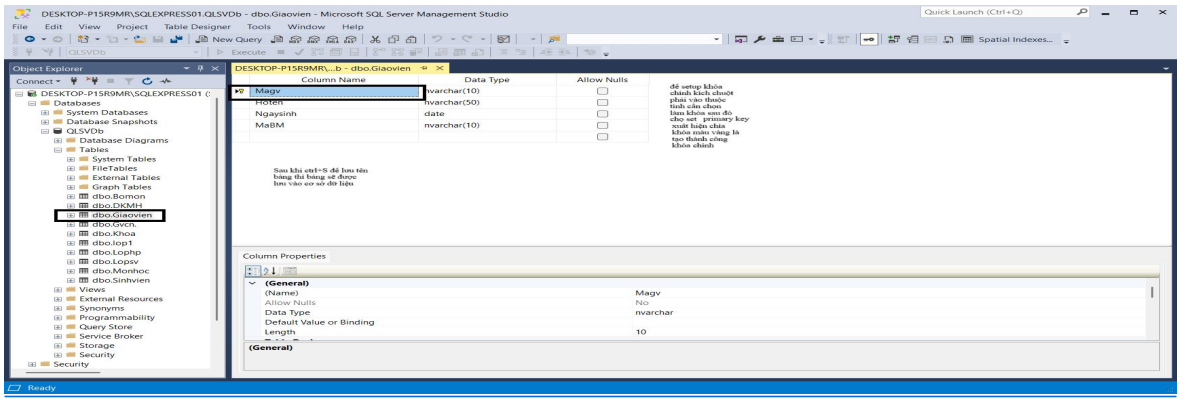
Sau khi tạo csdl nó sẽ xuất hiện bên trái giao diện sql tiếp đó kích chuột phải chọn tables->chuột phải tiếp chọn new->chọn table để tạo các bảng



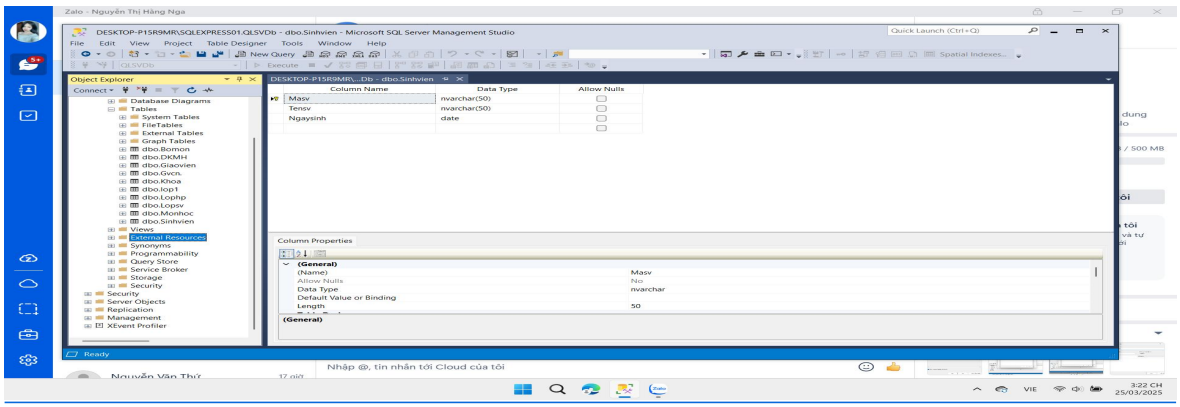
Xuất hiện giao diện tạo bảng nhập các thuộc tính và kiểu dữ liệu của bảng rồi ctrl+S->nhập tên bảng->chọn ok để lưu tên bảng vd như bảng giaovien

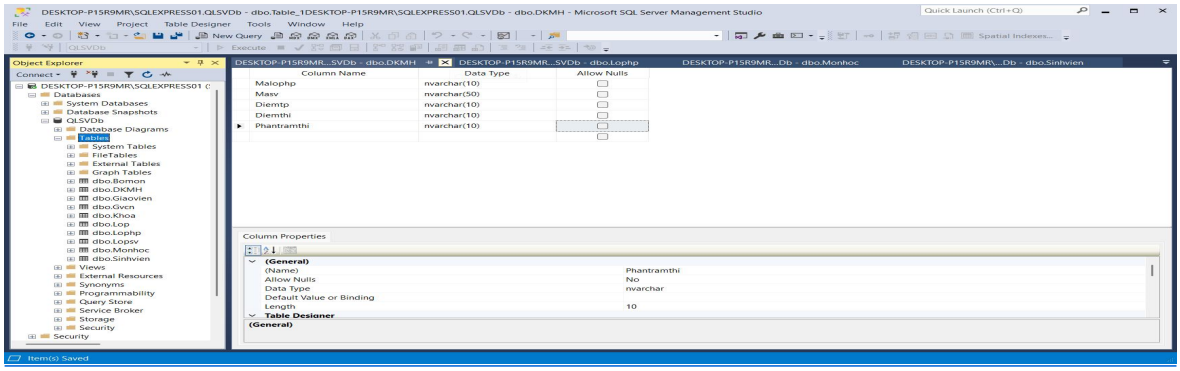


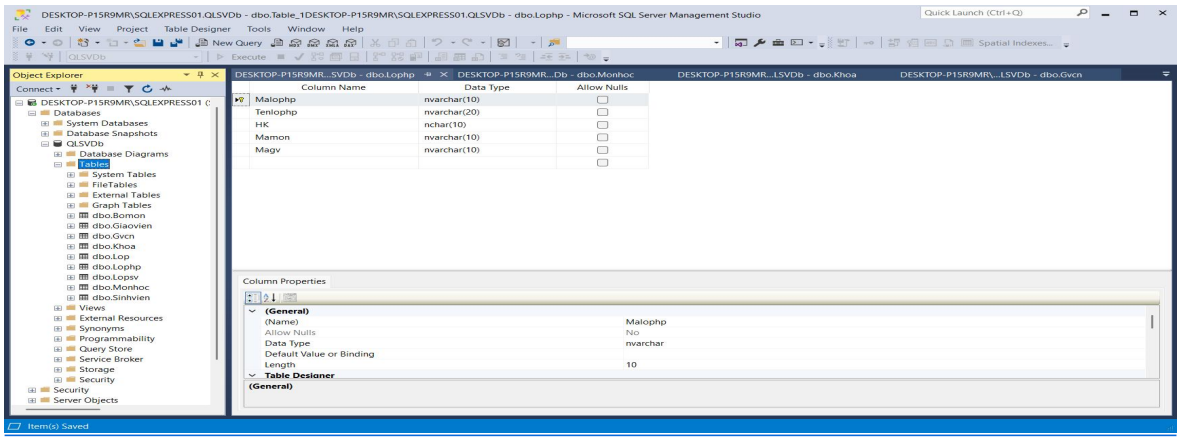


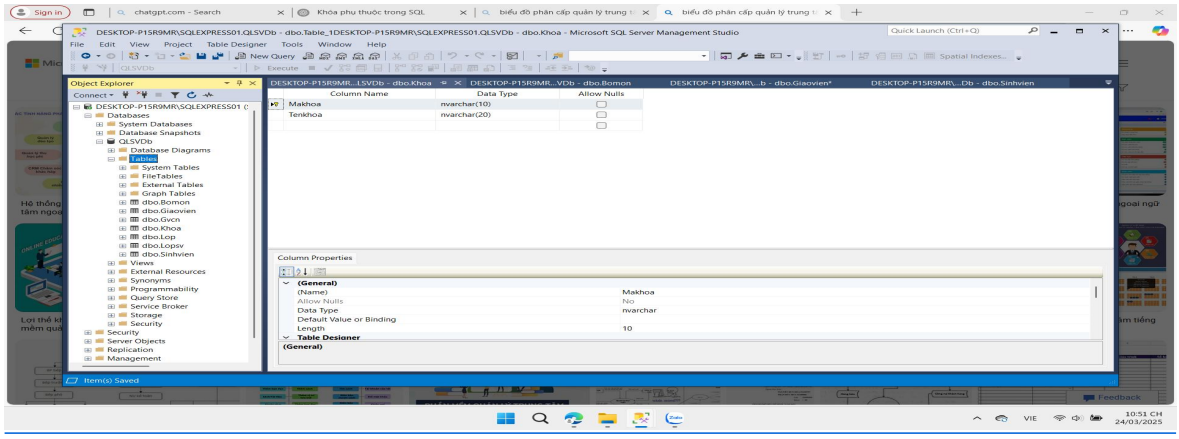


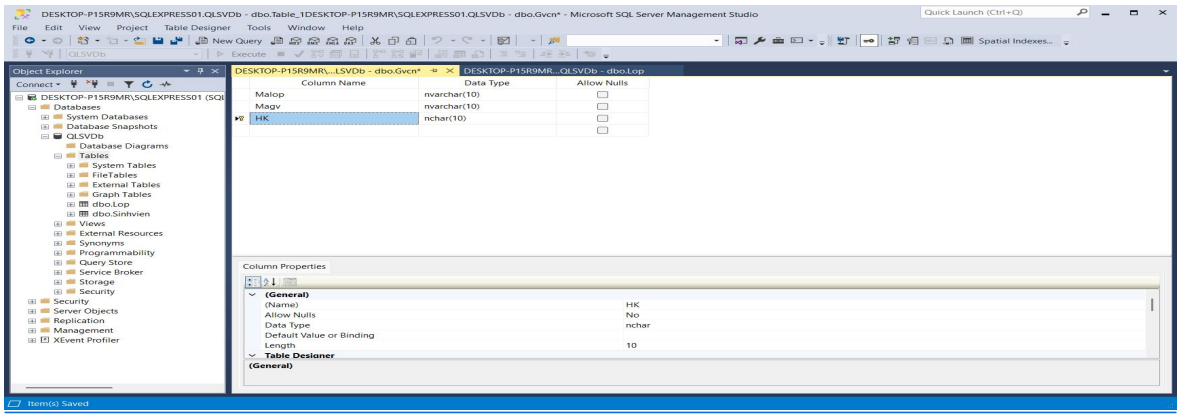
Tương tự cũng làm như vậy để tạo các bảng theo yêu cầu đề bài :Sinh vien,Lop,Gvcn,Lopsv,Bomon,Khoa,Monhoc,Lophp,DKMH bên dưới đây là các hình ảnh minh họa

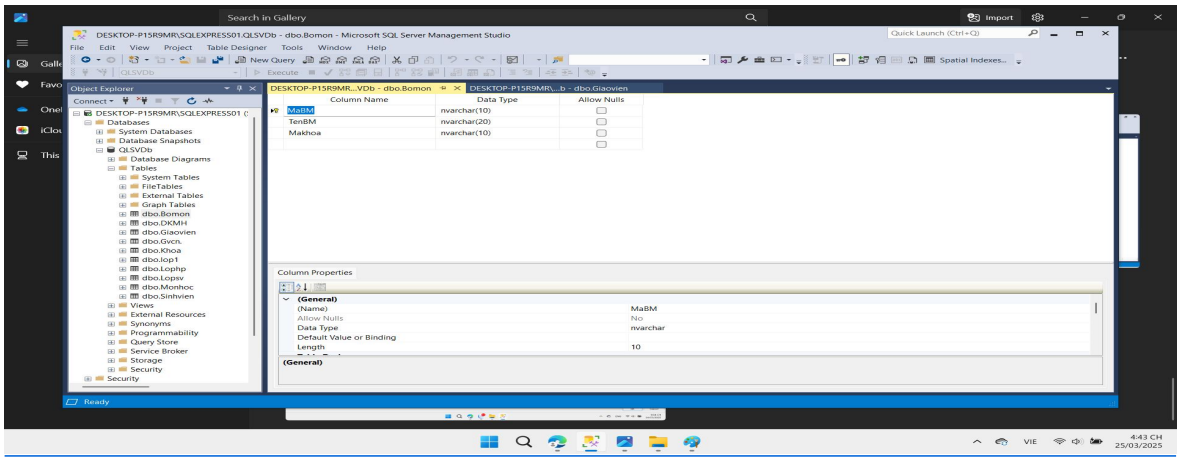


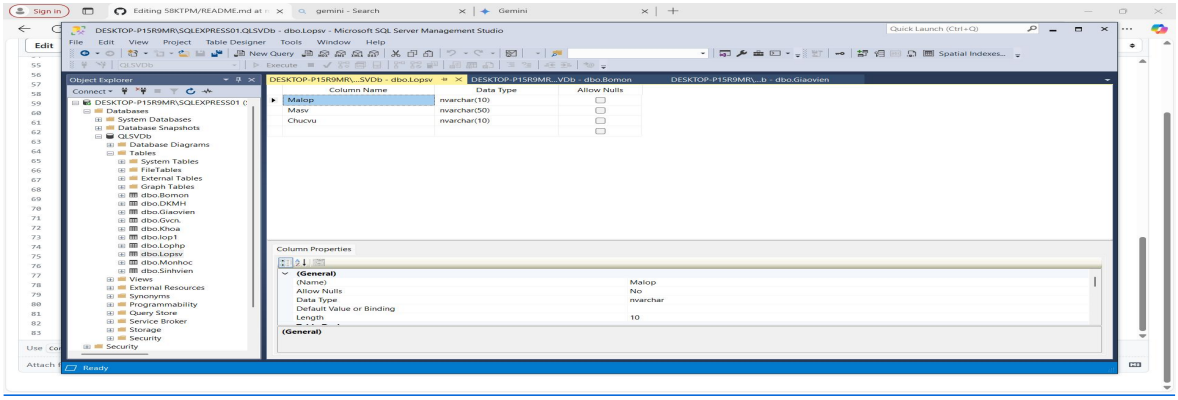


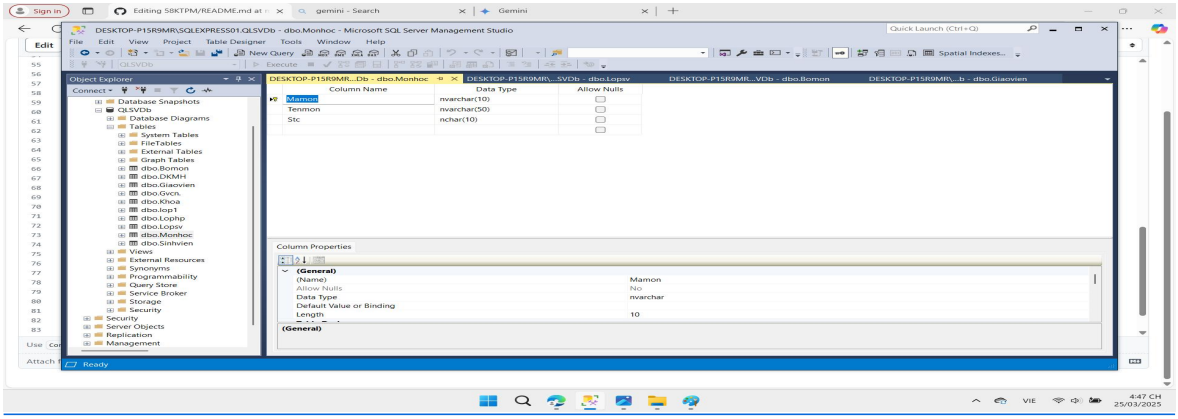

DESKTOP-P15R9MR\SQLEXPRESS01.QLSVDb - dbo.Table_1DESKTOP-P15R9MR\SQLEXPRESS01.QLSVDb - dbo.Lophp - Microsoft SQL Server Management Studio
Column Name
Data Type
Allow Nulls
Malophp nvarchar(10)
Tenlophp nvarchar(20)
HK nchar(10)
Mamon nvarchar(10)
Magv nvarchar(10)
Column Properties
(General)
(Name) Malophp
Allow Nulls No
Data Type nvarchar
Default Value or Binding
Length 10
Table Designer
(General)
Item(s) Saved


DESKTOP-P15R9MR\SQLEXPRESS01.QLSVDb - dbo.Table_1DESKTOP-P15R9MR\SQLEXPRESS01.QLSVDb - dbo.Khoa - Microsoft SQL Server Management Studio
Column Name
Data Type
Allow Nulls
Makhoa nvarchar(10)
Tenkhoa nvarchar(20)
Column Properties
(General)
(Name) Makhoa
Allow Nulls No
Data Type nvarchar
Default Value or Binding
Length 10
Table Designer
(General)
Item(s) Saved

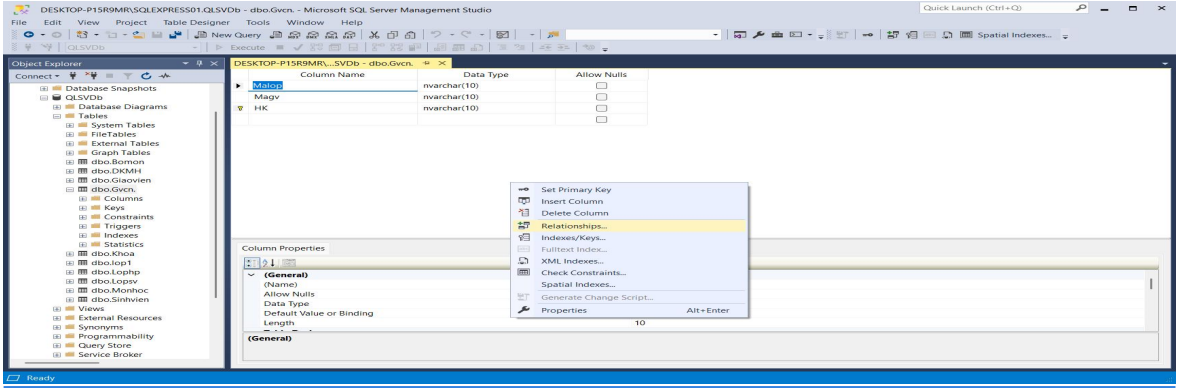

DESKTOP-P15R9MR\SQLEXPRESS01.QLSVDb - dbo.Table_1DESKTOP-P15R9MR\SQLEXPRESS01.QLSVDb - dbo.Gvcn* - Microsoft SQL Server Management Studio
Column Name
Data Type
Allow Nulls
Malop nvarchar(10)
Magv nvarchar(10)
HK nchar(10)
Column Properties
(General)
(Name) HK
Allow Nulls No
Data Type nchar
Default Value or Binding
Length 10
Table Designer
(General)
Item(s) Saved


DESKTOP-P15R9MR\QLSVDb - dbo.Bomon - Microsoft SQL Server Management Studio
Column Name
Data Type
Allow Nulls
MaBM nvarchar(10)
TenBM nvarchar(20)
Makhoa nvarchar(10)
Column Properties
(General)
(Name) MaBM
Allow Nulls No
Data Type nvarchar
Default Value or Binding
Length 10
(General)
Ready

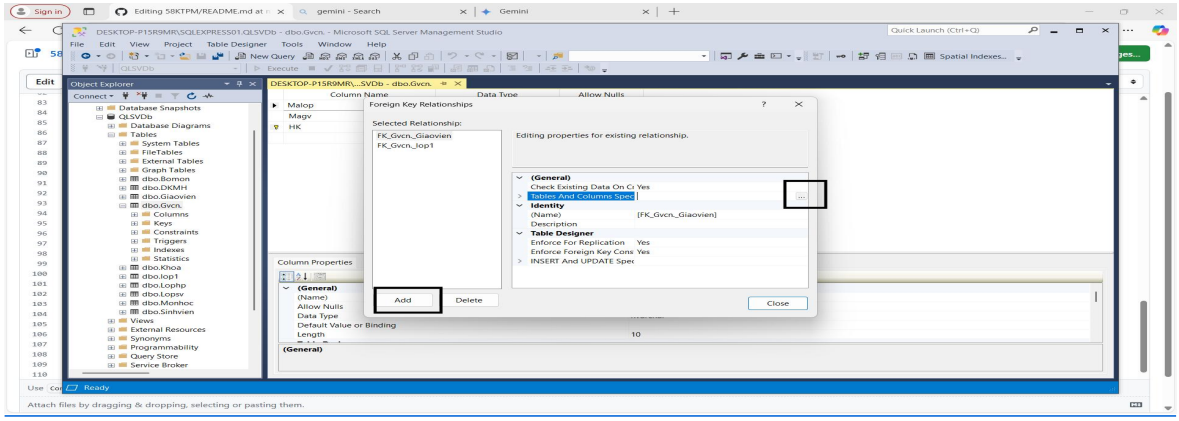




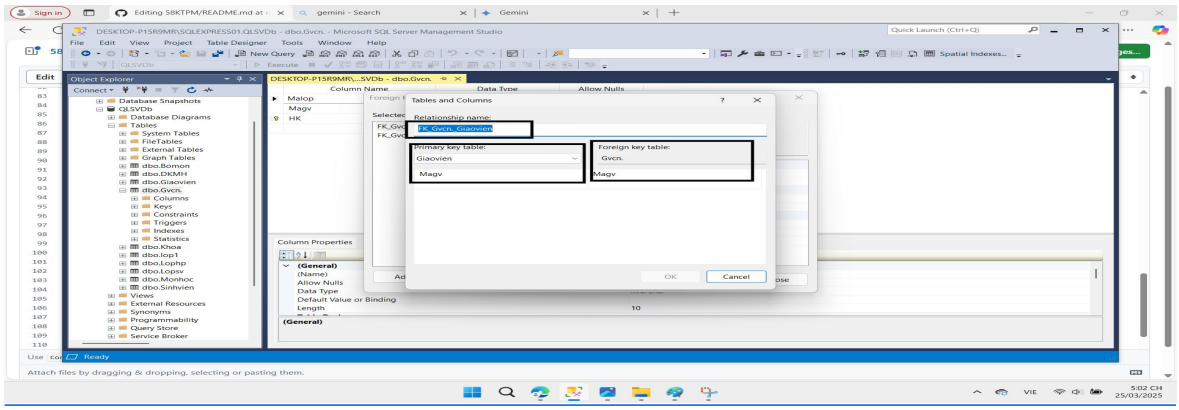
Để tạo khóa ngoại ta thực hiện các bước sau : Trongcơ sở dữ liệu nhấp chuột phải vào bảng mà bạn muốn tạo khóa ngoại (bảng con)->Chọn Relationships



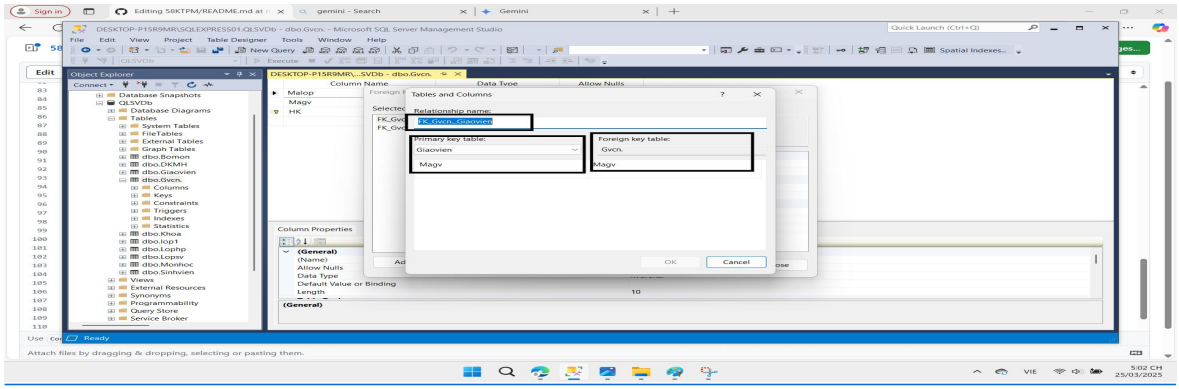
xuất hiện hộp thoại Foreign Key Relationships-> nhấp vào Add .Trong phần Tables and Columns Specification-> nhấp vào nút dấu ba chấm (...) bên cạnh "Tables and Columns Specification.



Trong hộp thoại "Tables and Columns", chọn bảng chính và cột khóa chính tương ứng.

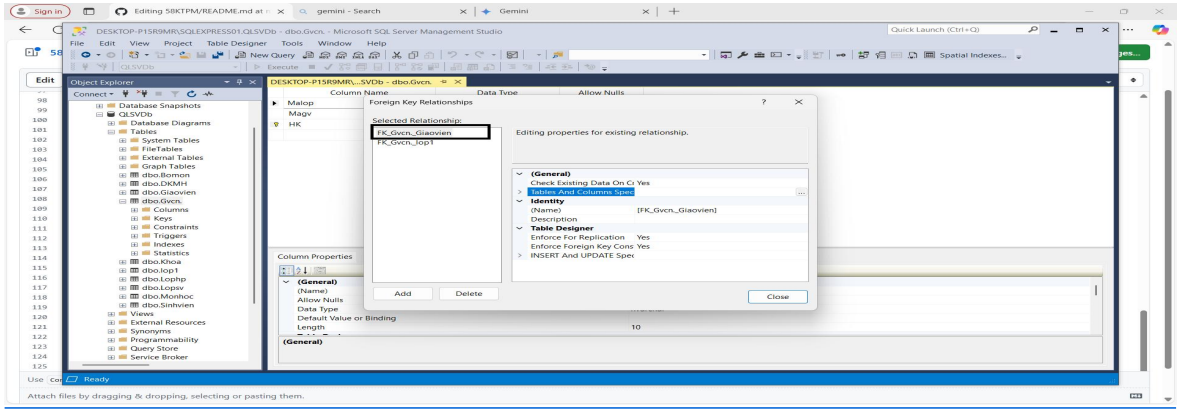


Trong đó Primary key table là bảng chính với Magv là khóa chính và foreign key table là bảng tham chiếu với Magv làm khóa ngoại



Lưu ý :Đảm bảo rằng kiểu dữ liệu của cột khóa ngoại trong bảng con phải khớp với kiểu dữ liệu của cột khóa chính trong bảng chính ví dụ cả hai đều cùng nvarchar(10)

Tiếp theo nhấp "OK" để đóng hộp thoại "Tables and Columns".

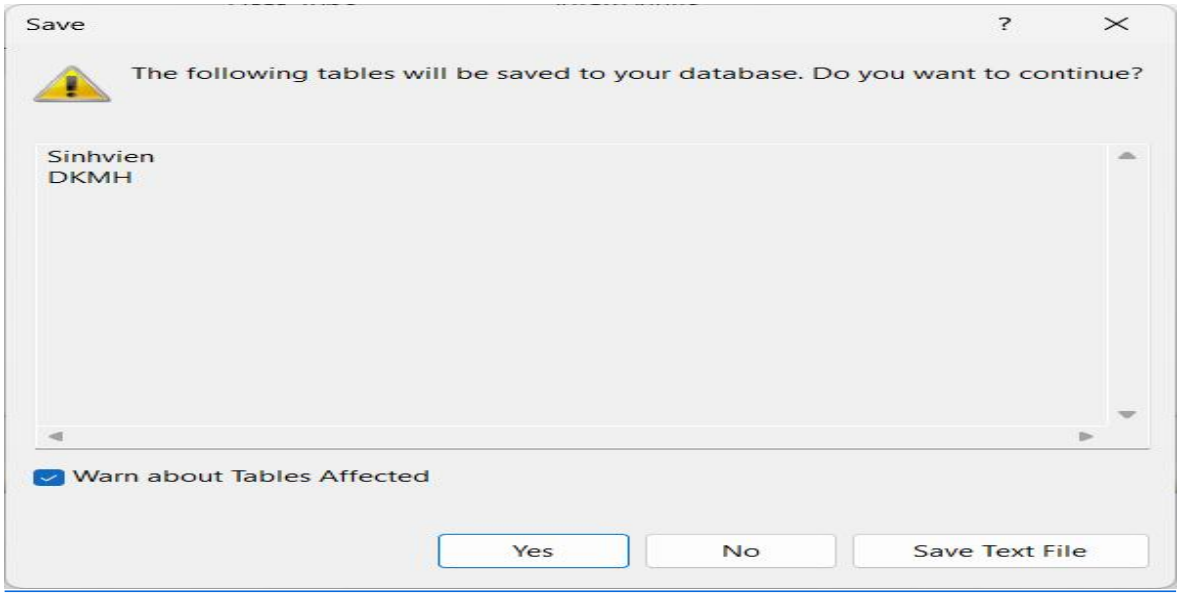


Tên của ràng buộc khóa ngoại là Foreign Key Constraint

Trong hộp thoại "Foreign Key Relationships" có thể đặt tên cho mối quan hệ khóa ngoại, chọn các quy tắc cập nhật và xóa

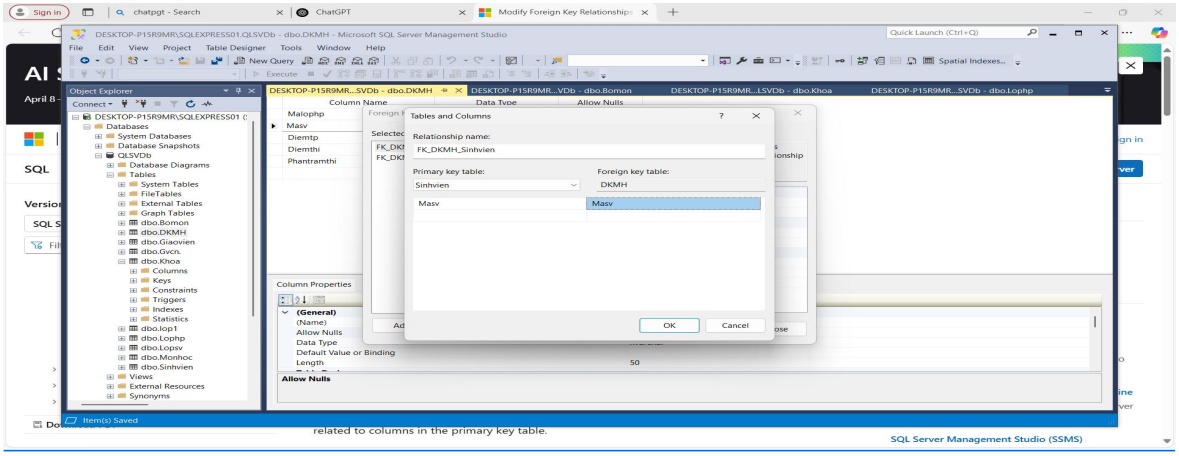
Nhấp "Close" để đóng hộp thoại "Foreign Key Relationships".

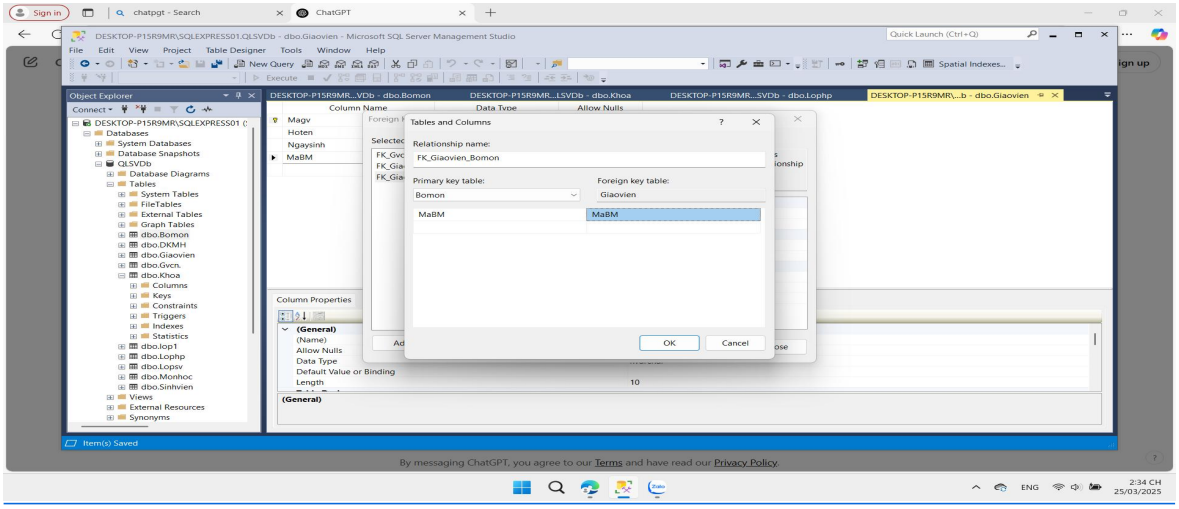
Cuối cùng nhấp vào nút "Save" trên thanh công cụ hoặc nhấn Ctrl+S để lưu sơ đồ cơ sở dữ liệu giao diện tương tự sẽ xuất hiện

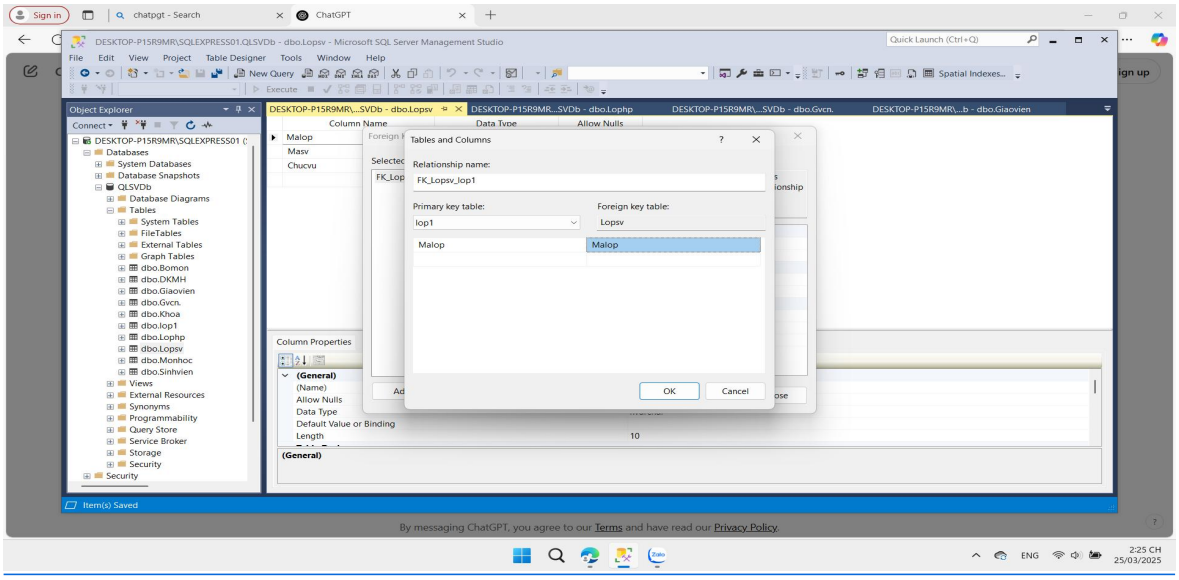


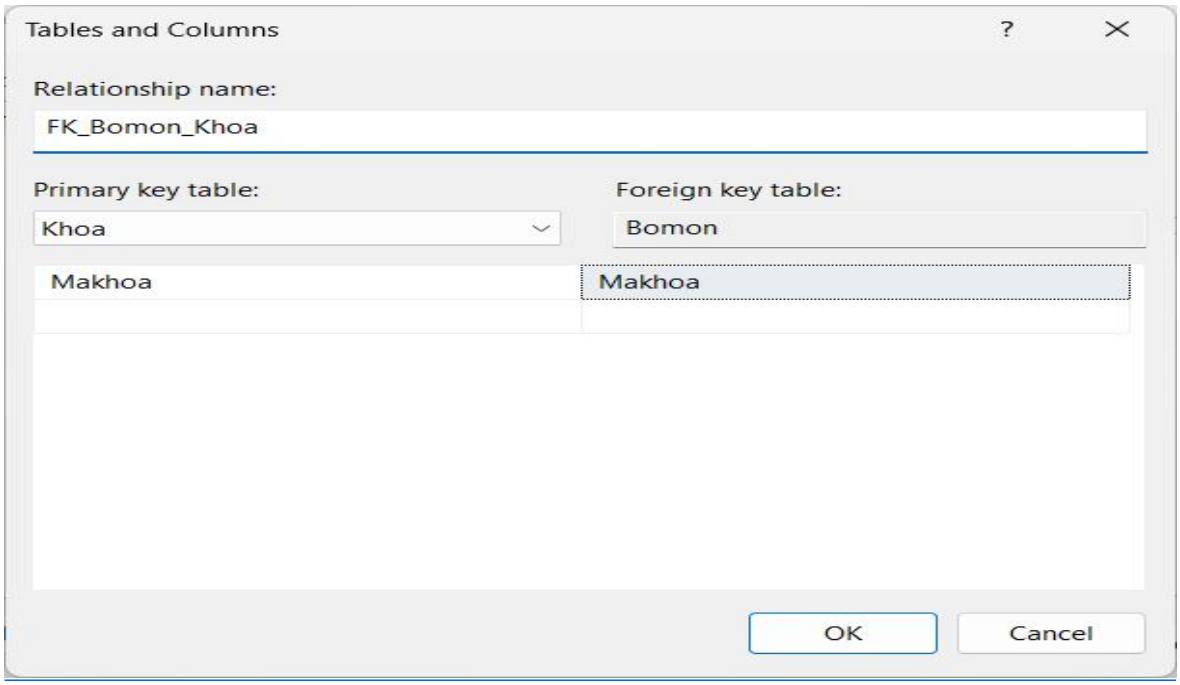
Đặt tên cho sơ đồ và nhấp "OK" để hoàn thành khóa ngoại

Tương tự như vậy đối với các khóa ngoại còn lại đều làm như vậy


Tables and Columns
Relationship name:
FK_DKMH_Sinhvien
Primary key table:
Sinhvien
Foreign key table:
DKMH
Masv
Masv
OK
Cancel


Tables and Columns
Relationship name:
FK_Giaovien_Bomon
Primary key table:
Bomon
Foreign key table:
Giaovien
MaBM
MaBM
OK
Cancel


Tables and Columns
Relationship name:
FK_Lopsv_Iop1
Primary key table:
Iop1
Foreign key table:
Lopsv
Malop
Malop
OK
Cancel



* Mã nguồn code sql :

```
USE [QLSVDb]
GO

/****** Object:  Table [dbo].[Sinhvien]    Script Date: 25/03/2025 2:40:01 CH ******/
SET ANSI_NULLS ON
GO
//taobangSinhvien
CREATE TABLE [dbo].[Sinhvien](
	[Masv] [nvarchar](50) NOT NULL,
	[Tensv] [nvarchar](50) NOT NULL,
	[Ngaysinh] [date] NOT NULL,
 CONSTRAINT [PK_Sinhvien] PRIMARY KEY
(
	[Masv] ASC)
GO
//taobanglop
CREATE TABLE [dbo].[lop1](
	[Malop] [nvarchar](10) NOT NULL,
	[Tenlop] [nvarchar](10) NOT NULL,
```

```
 CONSTRAINT [PK_lop1] PRIMARY KEY
(
        [Malop] ASC))
GO
//taobangGvcn
CREATE TABLE [dbo].[Gvcn.](
        [Malop] [nvarchar](10) NOT NULL,
        [Magv] [nvarchar](10) NOT NULL,
        [HK] [nvarchar](10) NOT NULL,
 CONSTRAINT [PK_Gvcn.] PRIMARY KEY
(
        [HK] ASC
)
GO
//taokhoangoai
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._lop1]
FOREIGN KEY([Malop])
REFERENCES [dbo].[lop1] ([Malop])
GO

ALTER TABLE [dbo].[Gvcn.] CHECK CONSTRAINT [FK_Gvcn._lop1]
GO
//taobanglopsv
CREATE TABLE [dbo].[Lopsv](
        [Malop] [nvarchar](10) NOT NULL,
        [Masv] [nvarchar](50) NOT NULL,
        [Chucvu] [nvarchar](10) NOT NULL
) ON [PRIMARY]
```

```
GO
//taokhoangoai
ALTER TABLE [dbo].[Lopsv]  WITH CHECK ADD  CONSTRAINT [FK_Lopsv_lop1]
FOREIGN KEY([Malop])
REFERENCES [dbo].[lop1] ([Malop])
GO
ALTER TABLE [dbo].[Lopsv] CHECK CONSTRAINT [FK_Lopsv_lop1]
GO
//taobanggiaovien
CREATE TABLE [dbo].[Giaovien](
	[Magv] [nvarchar](10) NOT NULL,
	[Hoten] [nvarchar](50) NOT NULL,
	[Ngaysinh] [date] NOT NULL,
	[MaBM] [nvarchar](10) NOT NULL,
 CONSTRAINT [PK_Giaovien] PRIMARY KEY
([Magv] ASC)
GO
//taokhoangoai
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
//taobangbomon
CREATE TABLE [dbo].[Bomon](
	[MaBM] [nvarchar](10) NOT NULL,
	[TenBM] [nvarchar](20) NOT NULL,
	[Makhoa] [nvarchar](10) NOT NULL,
 CONSTRAINT [PK_Bomon] PRIMARY KEY
([MaBM] ASC)
GO
//taokhoangoai
```

```
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
//taobangkhoa
CREATE TABLE [dbo].[Khoa](
	[Makhoa] [nvarchar](10) NOT NULL,
	[Tenkhoa] [nvarchar](20) NOT NULL,
 CONSTRAINT [PK_Khoa] PRIMARY KEY
([Makhoa] ASC)
GO
//taobangmonhoc
CREATE TABLE [dbo].[Monhoc](
	[Mamon] [nvarchar](10) NOT NULL,
	[Tenmon] [nvarchar](50) NOT NULL,
	[Stc] [nchar](10) NOT NULL,
 CONSTRAINT [PK_Monhoc] PRIMARY KEY
([Mamon] ASC)
GO
//taobanglophocphan
CREATE TABLE [dbo].[Lophp](
	[Malophp] [nvarchar](10) NOT NULL,
	[Tenlophp] [nvarchar](20) NOT NULL,
	[HK] [nchar](10) NOT NULL,
	[Mamon] [nvarchar](10) NOT NULL,
	[Magv] [nvarchar](10) NOT NULL,
 CONSTRAINT [PK_Lophp] PRIMARY KEY
([Malophp] ASC)
GO
//taokhoangoai
ALTER TABLE [dbo].[Lophp] WITH CHECK ADD CONSTRAINT
[FK_Lophp_Giaovien] FOREIGN KEY([Mamon])
```

```
REFERENCES [dbo].[Monhoc] ([Mamon])
GO
ALTER TABLE [dbo].[Lophp] CHECK CONSTRAINT [FK_Lophp_Giaovien]
GO
ALTER TABLE [dbo].[Lophp] WITH CHECK ADD CONSTRAINT
[FK_Lophp_Monhoc] FOREIGN KEY([Mamon])
REFERENCES [dbo].[Monhoc] ([Mamon])
GO
ALTER TABLE [dbo].[Lophp] CHECK CONSTRAINT [FK_Lophp_Monhoc]
//taobangDKMH
CREATE TABLE [dbo].[DKMH](
	[Malophp] [nvarchar](10) NOT NULL,
	[Masv] [nvarchar](50) NOT NULL,
	[Diemtp] [nvarchar](10) NOT NULL,
	[Diemthi] [nvarchar](10) NOT NULL,
	[Phantramthi] [nvarchar](10) NOT NULL
)
GO
//taokhoangoai
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
```

2.3 LINK TRA CỨU

1 Link github: 58KTPM/README.md at main · Nganguyen211/58KTPM

2 Mã qr :

CHƯƠNG 3 TRUY VẤN DỮ LIỆU

3.1 YÊU CẦU

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)

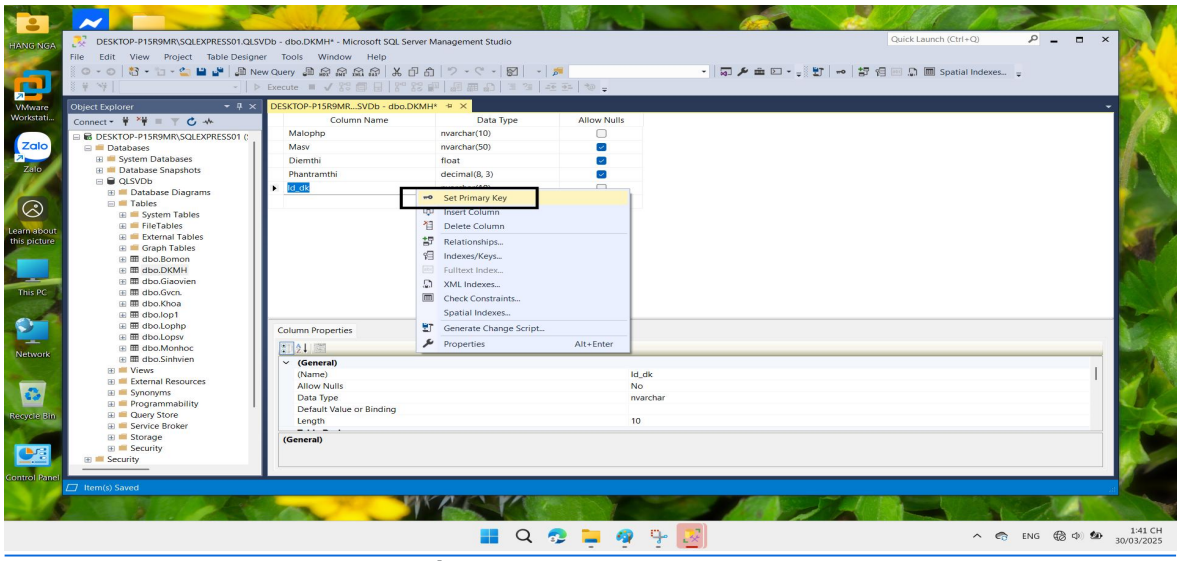
YÊU CẦU:

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.

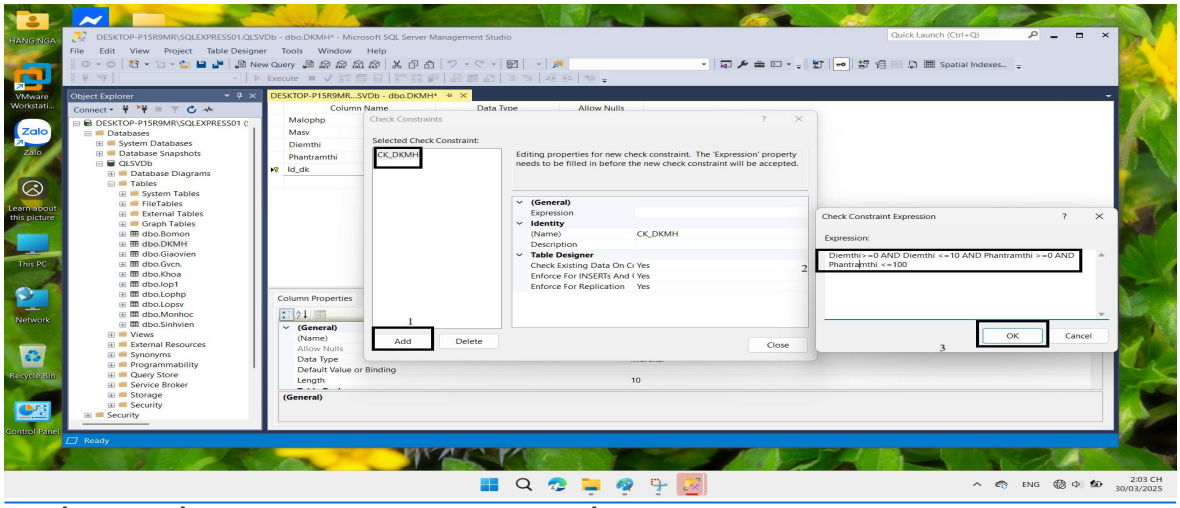
2.2 CÁC BƯỚC THỰC HIỆN

1. Sửa bảng DKMH và bảng Điểm từ bài tập 2 để có các bảng như yêu cầu

*)Sửa bảng DKMH theo yêu cầu:

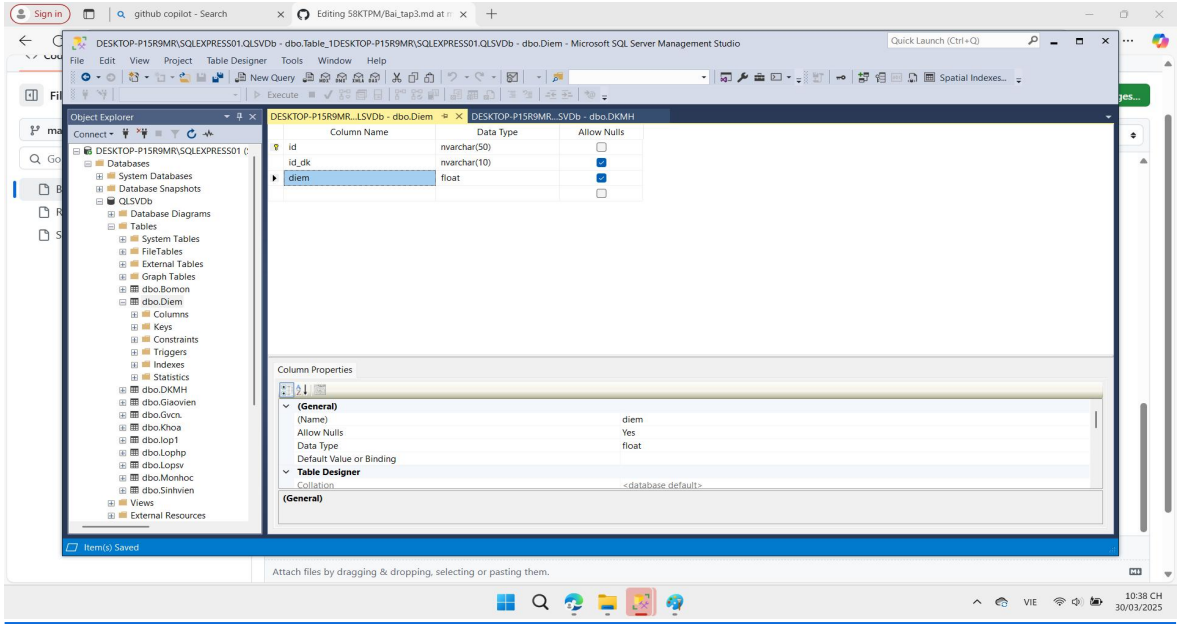


sửa bảng DKMH theo yêu cầu và set up khóa chính

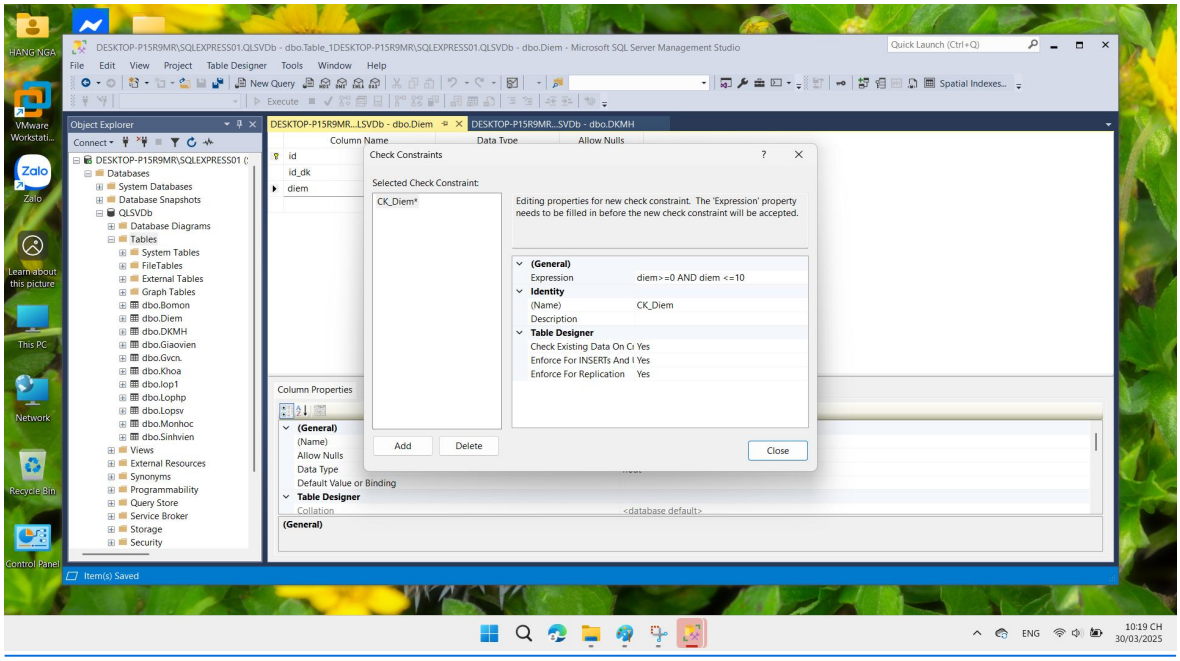


thiết lập điều kiện ,kích chuột phải xuất hiện hộp thoại và thực hiện các bước 1,2,3 như hình

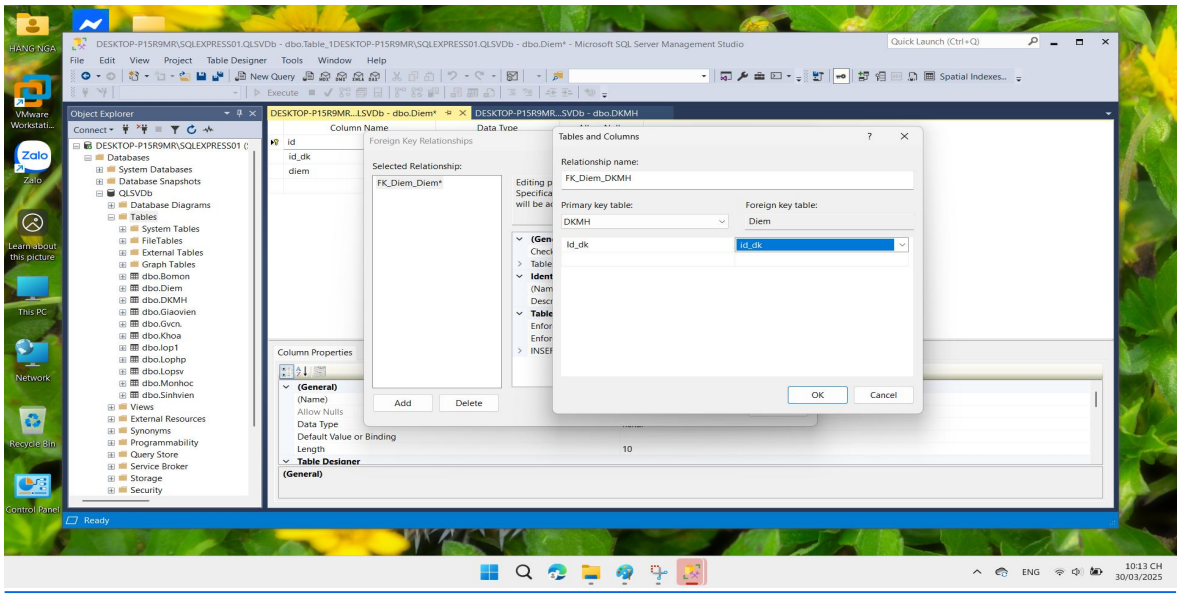
*)Tạo bảng điểm kích chuột phải vào tables chọn new chọn table để tạo bảng thêm các trường và thiết lập khóa chính



thiết lập điều kiện như bảng DKMH

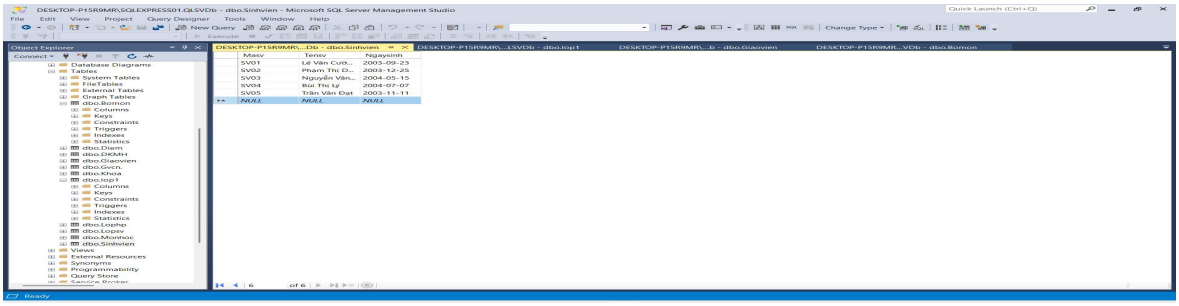


tạo khóa phụ liên kết giữa bảng điểm và DKMH như bt2

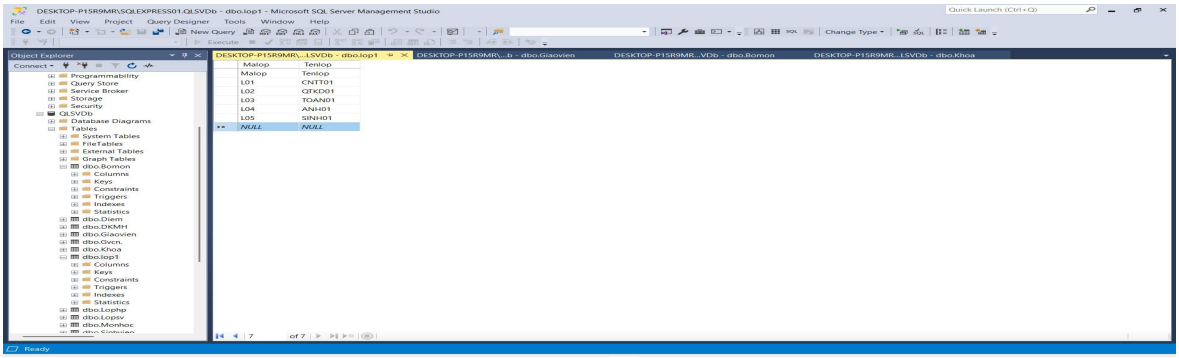


2. Nhập dữ liệu demo cho các bảng (nhập có kiểm soát từ tính năng Edit trên UI của mssm)

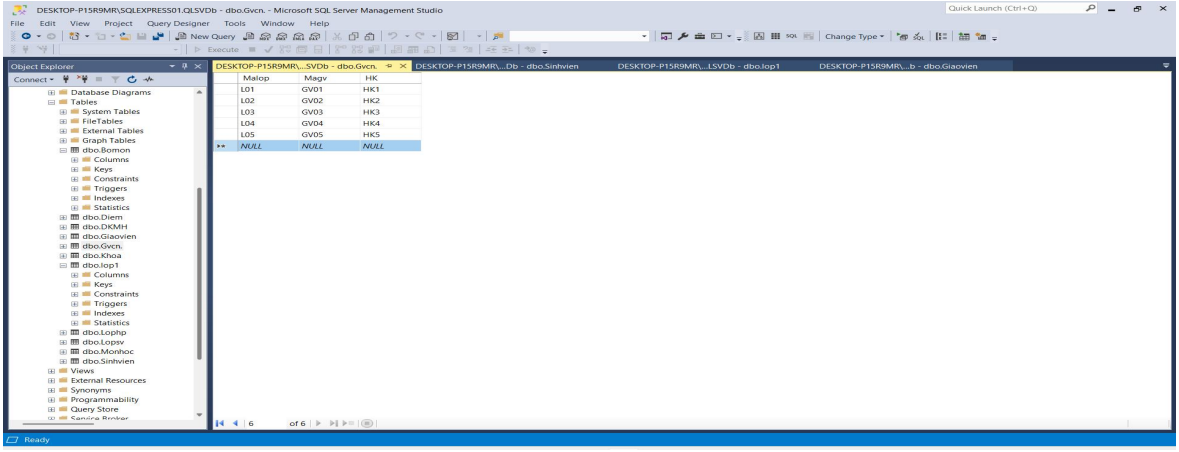
*)Dữ liệu demo bảng Sinhvien



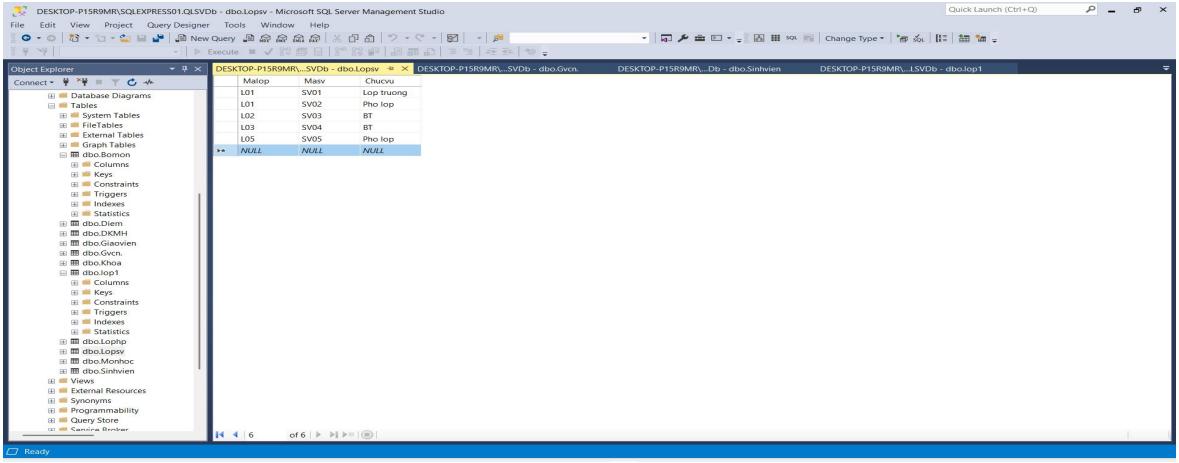
*)Dữ liệu demo bảng Lop(lop1): Đặt Unique cho tenLop để tránh sự trùng lặp cảnh báo khi hệ thống phát hiện sự trùng lặp



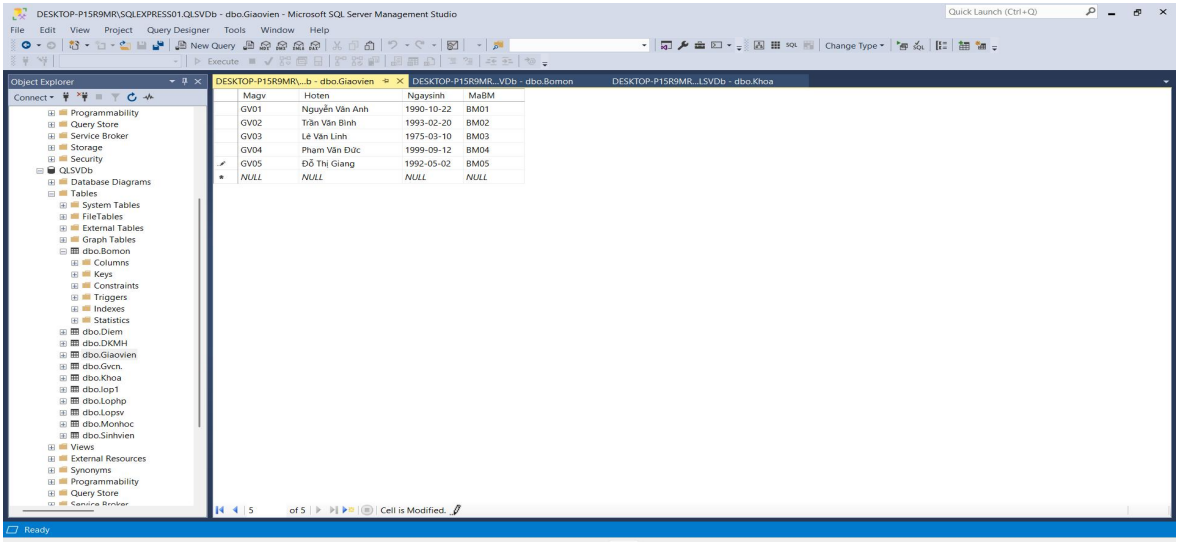
*)Dữ liệu demo bảng Gvcn



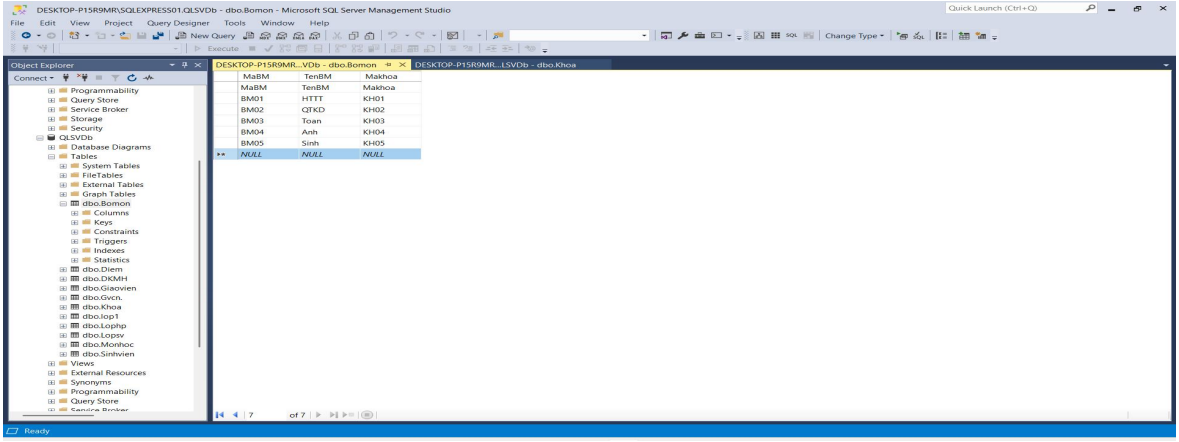
*)Dữ liệu demo bảng Lopsv



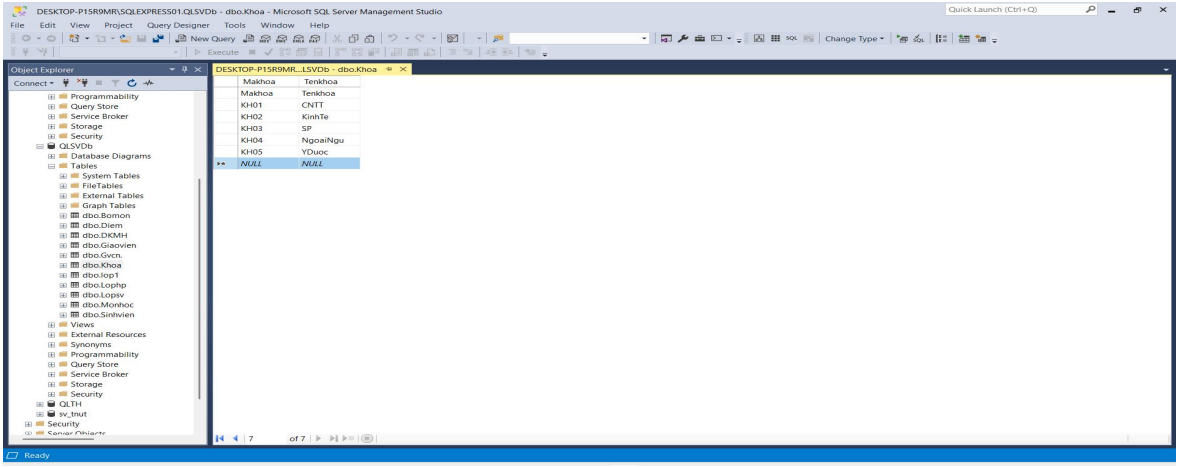
*)Dữ liệu demo bảng Giaovien



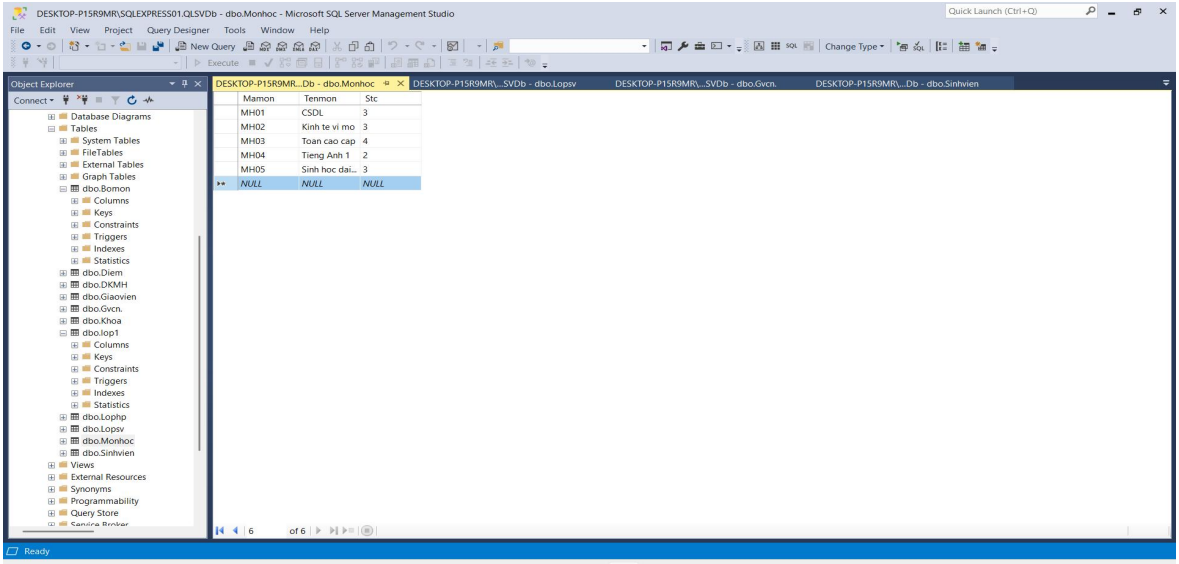
*)Dữ liệu demo bảng Bomon



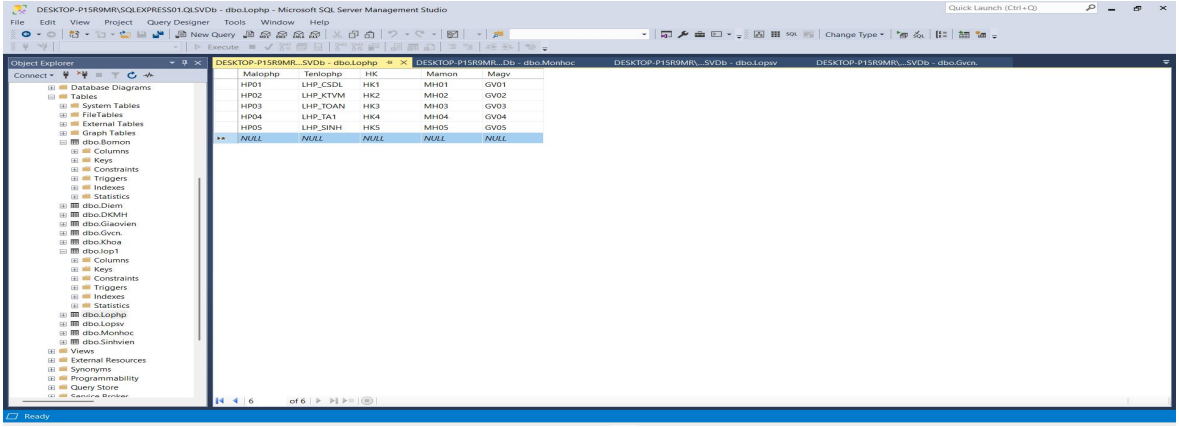
*)Dữ liệu demo bảng Khoa



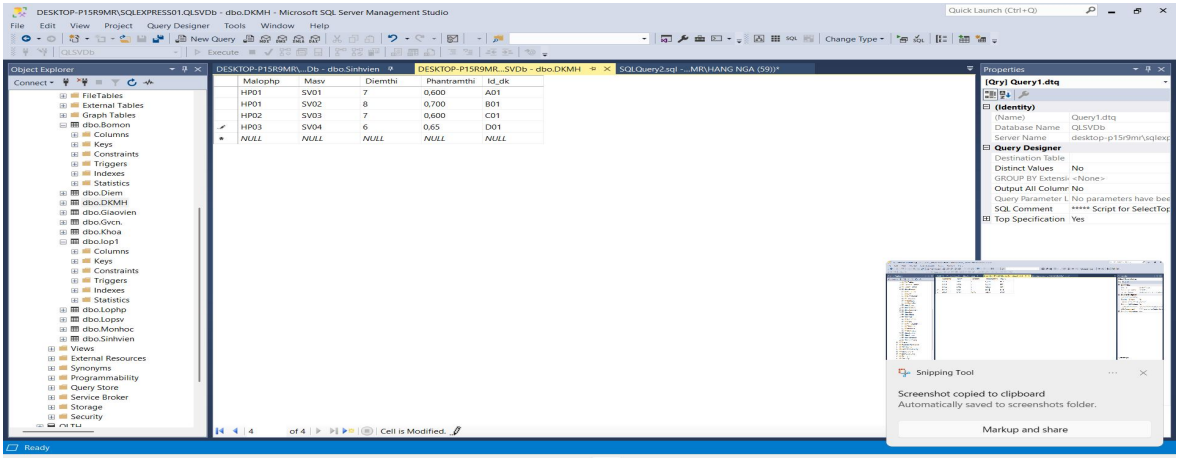
*)Dữ liệu demo bảng Monhoc



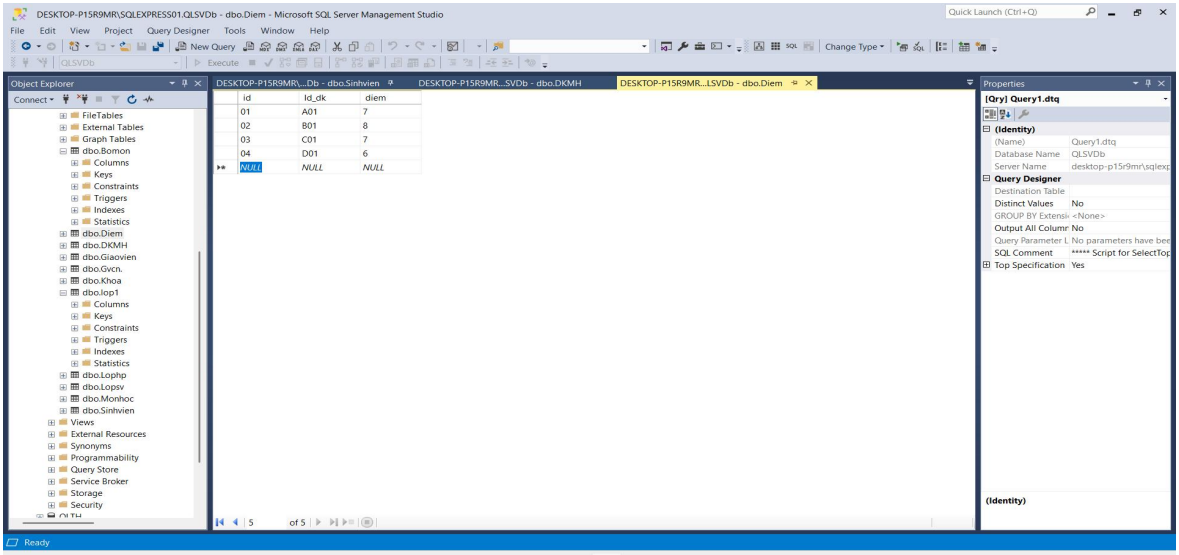
*)Dữ liệu demo bảng Lophp



*)Dữ liệu demo bảng DKMH

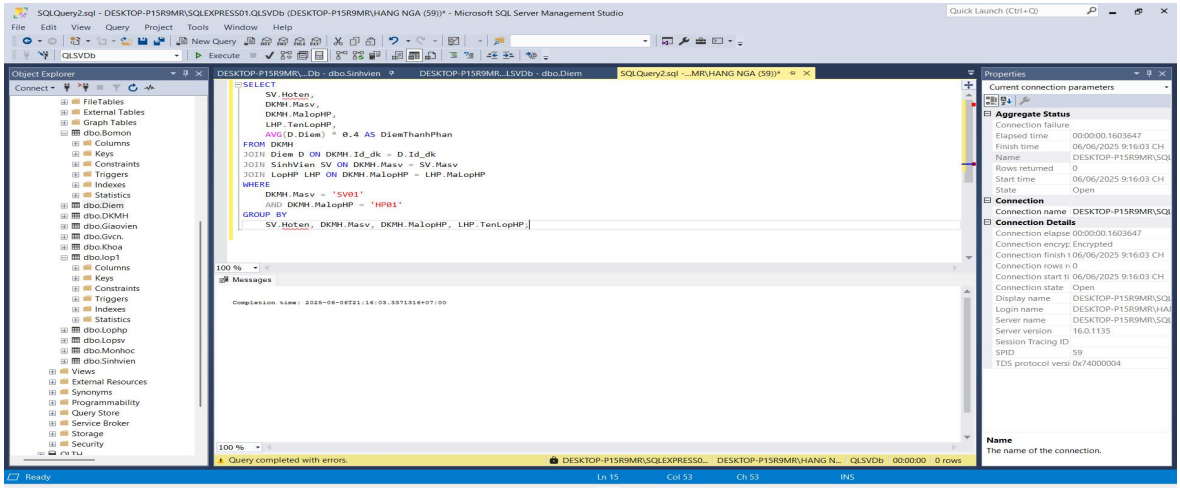


*)Dữ liệu demo bảng Diem

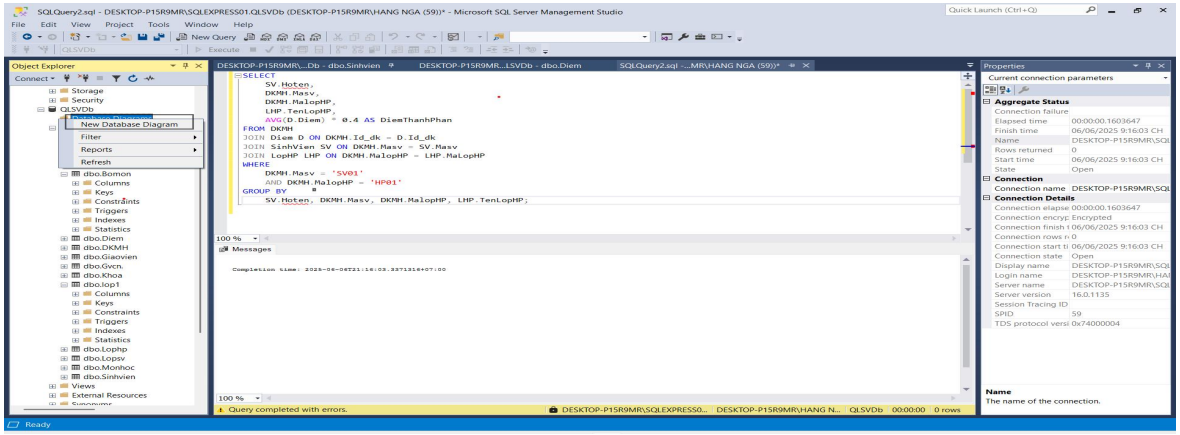


3.Viết lệnh truy vấn để: Tính được điểm thành phần của 1 sinh viên đang học tại 1 lớp học phần.

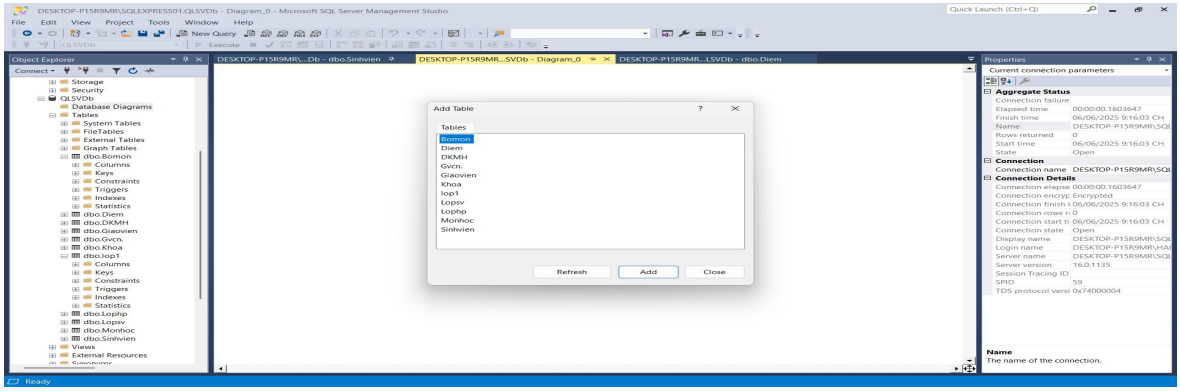
*)Lệnh truy vấn tính điểm thành phần của 1 sinh viên.



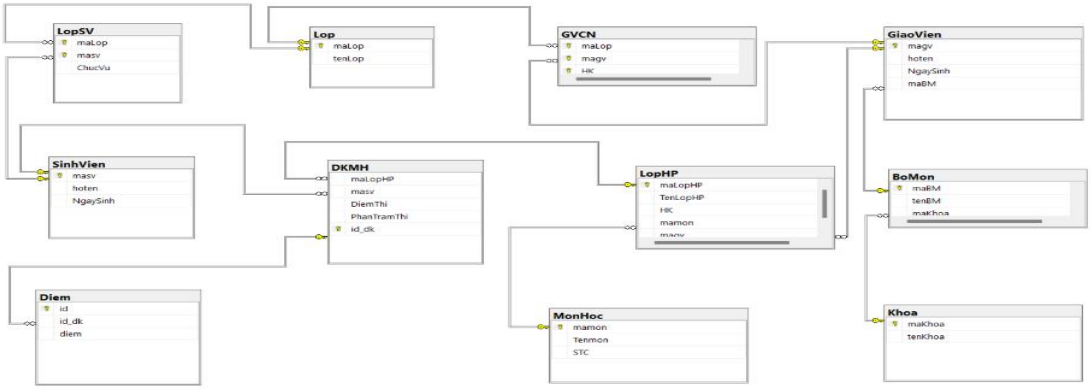
*)Thao tác tạo diagram mô tả các PK, FK của db.



Kích chuột phải vào database diagrams -> chọn new database diagram xuất hiện hộp thoại như bên dưới



Tiếp tục chọn tất cả các bảng và ấn Add kết quả được



* Mã code data sql

```
USE [QLSVDb]
GO
INSERT [dbo].[Sinhvien] ([Masv], [Tensv], [Ngaysinh]) VALUES (N'SV01', N'Lê Văn
Cường', CAST(N'2003-09-23' AS Date))
INSERT [dbo].[Sinhvien] ([Masv], [Tensv], [Ngaysinh]) VALUES (N'SV02', N'Phạm Thị
Dung', CAST(N'2003-12-25' AS Date))
INSERT [dbo].[Sinhvien] ([Masv], [Tensv], [Ngaysinh]) VALUES (N'SV03', N'Nguyễn Văn
Hà', CAST(N'2004-05-15' AS Date))
INSERT [dbo].[Sinhvien] ([Masv], [Tensv], [Ngaysinh]) VALUES (N'SV04', N'Bùi Thị Lý',
CAST(N'2004-07-07' AS Date))
INSERT [dbo].[Sinhvien] ([Masv], [Tensv], [Ngaysinh]) VALUES (N'SV05', N'Trần Văn
Đạt', CAST(N'2003-11-11' AS Date))
GO
INSERT [dbo].[Monhoc] ([Mamon], [Tenmon], [Stc]) VALUES (N'MH01', N'CSDL', N'3
')
INSERT [dbo].[Monhoc] ([Mamon], [Tenmon], [Stc]) VALUES (N'MH02', N'Kinh te vi mo',
N'3         ')
INSERT [dbo].[Monhoc] ([Mamon], [Tenmon], [Stc]) VALUES (N'MH03', N'Toan cao cap',
N'4         ')
INSERT [dbo].[Monhoc] ([Mamon], [Tenmon], [Stc]) VALUES (N'MH04', N'Tieng Anh 1',
N'2         ')
INSERT [dbo].[Monhoc] ([Mamon], [Tenmon], [Stc]) VALUES (N'MH05', N'Sinh hoc dai
cuong', N'3         ')
GO
INSERT [dbo].[Lophp] ([Malophp], [Tenlophp], [HK], [Mamon], [Magv]) VALUES (N'HP01',
N'LHP_CSDL', N'HK1       ', N'MH01', N'GV01')
INSERT [dbo].[Lophp] ([Malophp], [Tenlophp], [HK], [Mamon], [Magv]) VALUES (N'HP02',
N'LHP_KTVM', N'HK2       ', N'MH02', N'GV02')
INSERT [dbo].[Lophp] ([Malophp], [Tenlophp], [HK], [Mamon], [Magv]) VALUES (N'HP03',
N'LHP_TOAN', N'HK3       ', N'MH03', N'GV03')
INSERT [dbo].[Lophp] ([Malophp], [Tenlophp], [HK], [Mamon], [Magv]) VALUES (N'HP04',
N'LHP_TA1', N'HK4       ', N'MH04', N'GV04')
INSERT [dbo].[Lophp] ([Malophp], [Tenlophp], [HK], [Mamon], [Magv]) VALUES (N'HP05',
N'LHP_SINH', N'HK5       ', N'MH05', N'GV05')
GO
INSERT [dbo].[DKMH] ([Malophp], [Masv], [Diemthi], [Phantramthi], [Id_dk]) VALUES
(N'HP01', N'SV01', 7, CAST(0.600 AS Decimal(8, 3)), N'A01       ')
INSERT [dbo].[DKMH] ([Malophp], [Masv], [Diemthi], [Phantramthi], [Id_dk]) VALUES
(N'HP01', N'SV02', 8, CAST(0.700 AS Decimal(8, 3)), N'B01       ')
INSERT [dbo].[DKMH] ([Malophp], [Masv], [Diemthi], [Phantramthi], [Id_dk]) VALUES
(N'HP02', N'SV03', 7, CAST(0.600 AS Decimal(8, 3)), N'C01       ')
INSERT [dbo].[DKMH] ([Malophp], [Masv], [Diemthi], [Phantramthi], [Id_dk]) VALUES
(N'HP03', N'SV04', 6, CAST(65.000 AS Decimal(8, 3)), N'D01       ')
GO
```

```
INSERT [dbo].[Khoa] ([Makhoa], [Tenkhoa]) VALUES (N'KH01', N'CNTT')
INSERT [dbo].[Khoa] ([Makhoa], [Tenkhoa]) VALUES (N'KH02', N'KinhTe')
INSERT [dbo].[Khoa] ([Makhoa], [Tenkhoa]) VALUES (N'KH03', N'SP')
INSERT [dbo].[Khoa] ([Makhoa], [Tenkhoa]) VALUES (N'KH04', N'NgoaiNgu')
INSERT [dbo].[Khoa] ([Makhoa], [Tenkhoa]) VALUES (N'KH05', N'YDuoc')
INSERT [dbo].[Khoa] ([Makhoa], [Tenkhoa]) VALUES (N'Makhoa', N'Tenkhoa')
GO
INSERT [dbo].[Bomon] ([MaBM], [TenBM], [Makhoa]) VALUES (N'BM01', N'HTTT', N'KH01')
INSERT [dbo].[Bomon] ([MaBM], [TenBM], [Makhoa]) VALUES (N'BM02', N'QTKD', N'KH02')
INSERT [dbo].[Bomon] ([MaBM], [TenBM], [Makhoa]) VALUES (N'BM03', N'Toan', N'KH03')
INSERT [dbo].[Bomon] ([MaBM], [TenBM], [Makhoa]) VALUES (N'BM04', N'Anh', N'KH04')
INSERT [dbo].[Bomon] ([MaBM], [TenBM], [Makhoa]) VALUES (N'BM05', N'Sinh', N'KH05')
INSERT [dbo].[Bomon] ([MaBM], [TenBM], [Makhoa]) VALUES (N'MaBM', N'TenBM',
N'Makhoa')
GO
INSERT [dbo].[Giaovien] ([Magv], [Hoten], [Ngaysinh], [MaBM]) VALUES (N'GV01',
N'Nguyễn Vân Anh', CAST(N'1990-10-22' AS Date), N'BM01')
INSERT [dbo].[Giaovien] ([Magv], [Hoten], [Ngaysinh], [MaBM]) VALUES (N'GV02',
N'Trần Văn Bình', CAST(N'1993-02-20' AS Date), N'BM02')
INSERT [dbo].[Giaovien] ([Magv], [Hoten], [Ngaysinh], [MaBM]) VALUES (N'GV03', N'Lê
Văn Linh', CAST(N'1975-03-10' AS Date), N'BM03')
INSERT [dbo].[Giaovien] ([Magv], [Hoten], [Ngaysinh], [MaBM]) VALUES (N'GV04',
N'Phạm Văn Đức', CAST(N'1999-09-12' AS Date), N'BM04')
INSERT [dbo].[Giaovien] ([Magv], [Hoten], [Ngaysinh], [MaBM]) VALUES (N'GV05', N'Đỗ
Thị Giang', CAST(N'2004-05-02' AS Date), N'BM05')
GO
INSERT [dbo].[lop1] ([Malop], [Tenlop]) VALUES (N'L01', N'CNTT01')
INSERT [dbo].[lop1] ([Malop], [Tenlop]) VALUES (N'L02', N'QTKD01')
INSERT [dbo].[lop1] ([Malop], [Tenlop]) VALUES (N'L03', N'TOAN01')
INSERT [dbo].[lop1] ([Malop], [Tenlop]) VALUES (N'L04', N'ANH01')
INSERT [dbo].[lop1] ([Malop], [Tenlop]) VALUES (N'L05', N'SINH01')
INSERT [dbo].[lop1] ([Malop], [Tenlop]) VALUES (N'Malop', N'Tenlop')
GO
INSERT [dbo].[Gvcn.] ([Malop], [Magv], [HK]) VALUES (N'L01', N'GV01', N'HK1')
INSERT [dbo].[Gvcn.] ([Malop], [Magv], [HK]) VALUES (N'L02', N'GV02', N'HK2')
INSERT [dbo].[Gvcn.] ([Malop], [Magv], [HK]) VALUES (N'L03', N'GV03', N'HK3')
INSERT [dbo].[Gvcn.] ([Malop], [Magv], [HK]) VALUES (N'L04', N'GV04', N'HK4')
INSERT [dbo].[Gvcn.] ([Malop], [Magv], [HK]) VALUES (N'L05', N'GV05', N'HK5')
GO
INSERT [dbo].[Lopsv] ([Malop], [Masv], [Chucvu]) VALUES (N'L02', N'SV03', N'BT')
INSERT [dbo].[Lopsv] ([Malop], [Masv], [Chucvu]) VALUES (N'L01', N'SV01', N'Lop
truong')
INSERT [dbo].[Lopsv] ([Malop], [Masv], [Chucvu]) VALUES (N'L01', N'SV02', N'Pho lop')
INSERT [dbo].[Lopsv] ([Malop], [Masv], [Chucvu]) VALUES (N'L03', N'SV04', N'BT')
INSERT [dbo].[Lopsv] ([Malop], [Masv], [Chucvu]) VALUES (N'L05', N'SV05', N'Pho lop')
GO
INSERT [dbo].[Diem] ([id], [Id_dk], [diem]) VALUES (N'01', N'A01       ', 7)
INSERT [dbo].[Diem] ([id], [Id_dk], [diem]) VALUES (N'02', N'B01       ', 8)
INSERT [dbo].[Diem] ([id], [Id_dk], [diem]) VALUES (N'03', N'C01       ', 7)
INSERT [dbo].[Diem] ([id], [Id_dk], [diem]) VALUES (N'04', N'D01       ', 6)
GO
```

* Mã code schema sql

```
USE [QLSVDb]
GO
/****** Object:  Table [dbo].[Bomon]    Script Date: 06/06/2025 10:10:26 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Bomon](
```

```
	[MaBM] [nvarchar](10) NOT NULL,
	[TenBM] [nvarchar](20) NOT NULL,
	[Makhoa] [nvarchar](10) NOT NULL,
 CONSTRAINT [PK_Bomon] PRIMARY KEY CLUSTERED
(
	[MaBM] 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
/****** Object:  Table [dbo].[Diem]    Script Date: 06/06/2025 10:10:26 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Diem](
	[id] [nvarchar](50) NOT NULL,
	[Id_dk] [nchar](10) NULL,
	[diem] [float] NULL,
 CONSTRAINT [PK_Diem] PRIMARY KEY CLUSTERED
(
	[id] 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
/****** Object:  Table [dbo].[DKMH]    Script Date: 06/06/2025 10:10:26 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[DKMH](
	[Malophp] [nvarchar](10) NOT NULL,
	[Masv] [nvarchar](50) NULL,
	[Diemthi] [float] NULL,
	[Phantramthi] [decimal](8, 3) NULL,
	[Id_dk] [nchar](10) NOT NULL,
 CONSTRAINT [PK_DKMH] PRIMARY KEY CLUSTERED
(
	[Id_dk] 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
/****** Object:  Table [dbo].[Giaovien]    Script Date: 06/06/2025 10:10:26 CH
******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Giaovien](
	[Magv] [nvarchar](10) NOT NULL,
	[Hoten] [nvarchar](50) NOT NULL,
	[Ngaysinh] [date] NOT NULL,
	[MaBM] [nvarchar](10) NOT NULL,
 CONSTRAINT [PK_Giaovien] PRIMARY KEY CLUSTERED
(
	[Magv] 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
/****** Object:  Table [dbo].[Gvcn.]    Script Date: 06/06/2025 10:10:26 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Gvcn.](
	[Malop] [nvarchar](10) NOT NULL,
	[Magv] [nvarchar](10) NOT NULL,
	[HK] [nvarchar](10) NOT NULL,
 CONSTRAINT [PK_Gvcn.] PRIMARY KEY CLUSTERED
(
	[HK] 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
/****** Object:  Table [dbo].[Khoa]    Script Date: 06/06/2025 10:10:26 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Khoa](
	[Makhoa] [nvarchar](10) NOT NULL,
	[Tenkhoa] [nvarchar](20) NOT NULL,
 CONSTRAINT [PK_Khoa] PRIMARY KEY CLUSTERED
(
	[Makhoa] 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
/****** Object:  Table [dbo].[lop1]    Script Date: 06/06/2025 10:10:26 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[lop1](
	[Malop] [nvarchar](10) NOT NULL,
	[Tenlop] [nvarchar](10) NOT NULL,
 CONSTRAINT [PK_lop1] PRIMARY KEY CLUSTERED
(
	[Malop] 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
/****** Object:  Table [dbo].[Lophp]    Script Date: 06/06/2025 10:10:26 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Lophp](
	[Malophp] [nvarchar](10) NOT NULL,
	[Tenlophp] [nvarchar](20) NOT NULL,
	[HK] [nchar](10) NOT NULL,
```

```
	[Mamon] [nvarchar](10) NOT NULL,
	[Magv] [nvarchar](10) NOT NULL,
 CONSTRAINT [PK_Lophp] 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
/****** Object:  Table [dbo].[Lopsv]    Script Date: 06/06/2025 10:10:26 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Lopsv](
	[Malop] [nvarchar](10) NOT NULL,
	[Masv] [nvarchar](50) NOT NULL,
	[Chucvu] [nvarchar](10) NOT NULL,
 CONSTRAINT [UQ_Lopsv_Masv] UNIQUE NONCLUSTERED
(
	[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
/****** Object:  Table [dbo].[Monhoc]    Script Date: 06/06/2025 10:10:26 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Monhoc](
	[Mamon] [nvarchar](10) NOT NULL,
	[Tenmon] [nvarchar](50) NOT NULL,
	[Stc] [nchar](10) NOT NULL,
 CONSTRAINT [PK_Monhoc] PRIMARY KEY CLUSTERED
(
	[Mamon] 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
/****** Object:  Table [dbo].[Sinhvien]    Script Date: 06/06/2025 10:10:26 CH
******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Sinhvien](
	[Masv] [nvarchar](50) NOT NULL,
	[Tensv] [nvarchar](50) NOT NULL,
	[Ngaysinh] [date] NOT 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].[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
ALTER TABLE [dbo].[Diem]  WITH CHECK ADD  CONSTRAINT [FK_Diem_DKMH] FOREIGN
KEY([Id_dk])
REFERENCES [dbo].[DKMH] ([Id_dk])
GO
ALTER TABLE [dbo].[Diem] CHECK CONSTRAINT [FK_Diem_DKMH]
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
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
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._lop1] FOREIGN
KEY([Malop])
REFERENCES [dbo].[lop1] ([Malop])
GO
ALTER TABLE [dbo].[Gvcn.] CHECK CONSTRAINT [FK_Gvcn._lop1]
GO
ALTER TABLE [dbo].[Lophp]  WITH CHECK ADD  CONSTRAINT [FK_Lophp_Giaovien] FOREIGN
KEY([Mamon])
REFERENCES [dbo].[Monhoc] ([Mamon])
GO
ALTER TABLE [dbo].[Lophp] CHECK CONSTRAINT [FK_Lophp_Giaovien]
GO
ALTER TABLE [dbo].[Lophp]  WITH CHECK ADD  CONSTRAINT [FK_Lophp_Monhoc] FOREIGN
KEY([Mamon])
REFERENCES [dbo].[Monhoc] ([Mamon])
GO
ALTER TABLE [dbo].[Lophp] CHECK CONSTRAINT [FK_Lophp_Monhoc]
GO
ALTER TABLE [dbo].[Lopsv]  WITH CHECK ADD  CONSTRAINT [FK_Lopsv_lop1] FOREIGN
KEY([Malop])
REFERENCES [dbo].[lop1] ([Malop])
GO
ALTER TABLE [dbo].[Lopsv] CHECK CONSTRAINT [FK_Lopsv_lop1]
GO
ALTER TABLE [dbo].[Diem]  WITH CHECK ADD  CONSTRAINT [CK_Diem] CHECK  (([diem]>=(0)
AND [diem]<=(10)))
GO
```

```
ALTER TABLE [dbo].[Diem] CHECK CONSTRAINT [CK_Diem]
GO
ALTER TABLE [dbo].[DKMH]  WITH CHECK ADD  CONSTRAINT [CK_DKMH] CHECK
(([Diemthi]>=(0) AND [Diemthi]<=(10) AND [Phantramthi]>=(0) AND
[Phantramthi]<=(100)))
GO
ALTER TABLE [dbo].[DKMH] CHECK CONSTRAINT [CK_DKMH]
GO
```

3.3 LINK TRA CỨU

1 Link github: [58KTPM/README.md at main · Nganguyen211/58KTPM]

2 Mã qr :

CHƯƠNG 4 TẠO QUERY TRUY VẤN

4.1 YÊU CẦU

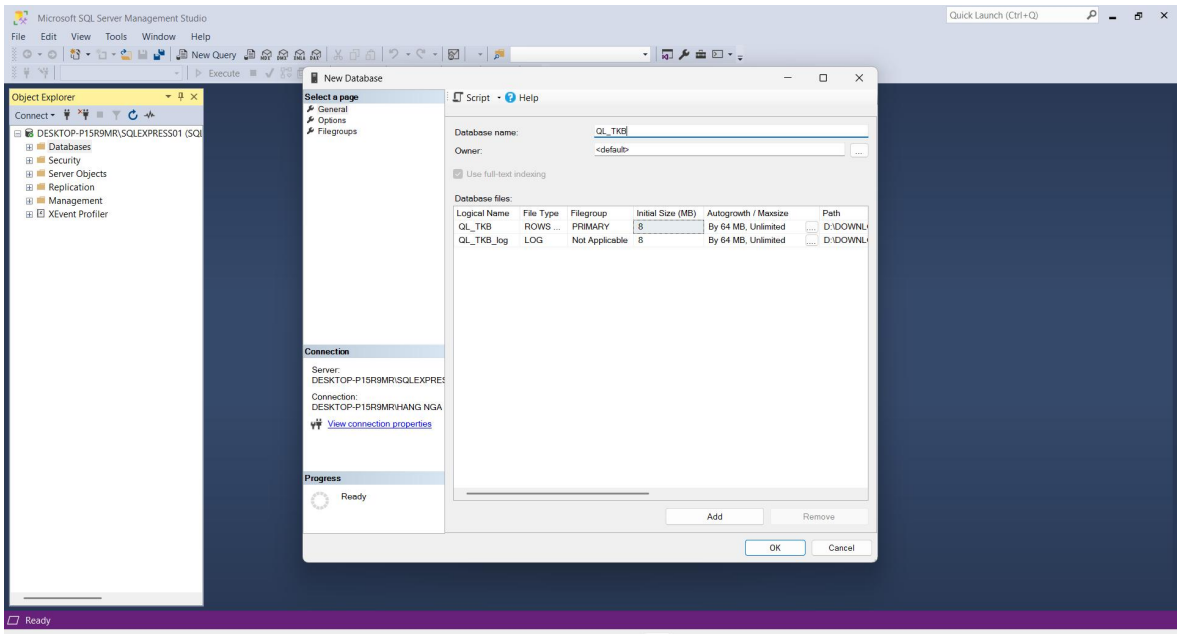
Bài toán:

- 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.

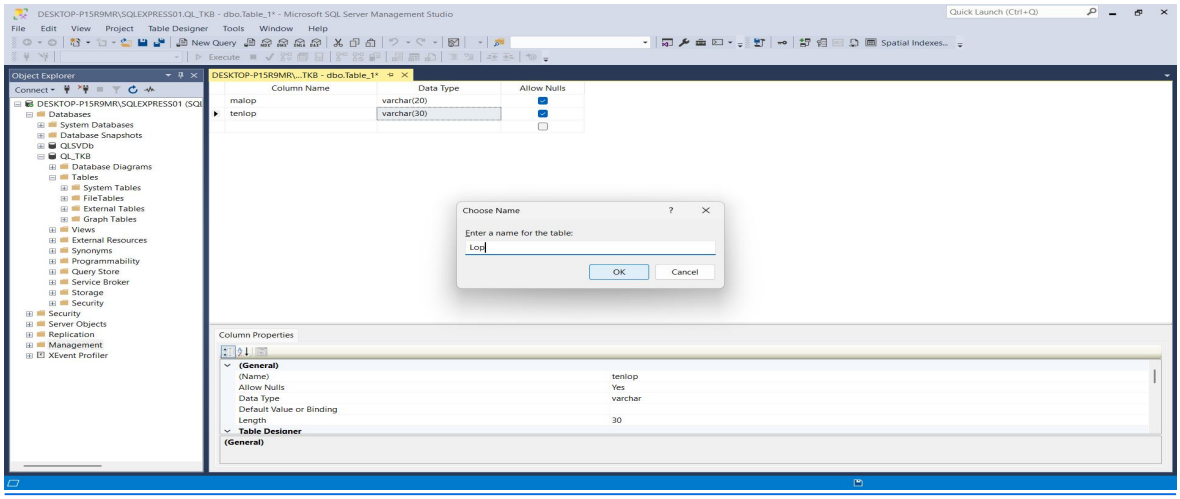
trả lời câu hỏi: trong khoảng thời gian từ datetime1 tới datetime2 thì có những gv nào đang bận giảng dạy.

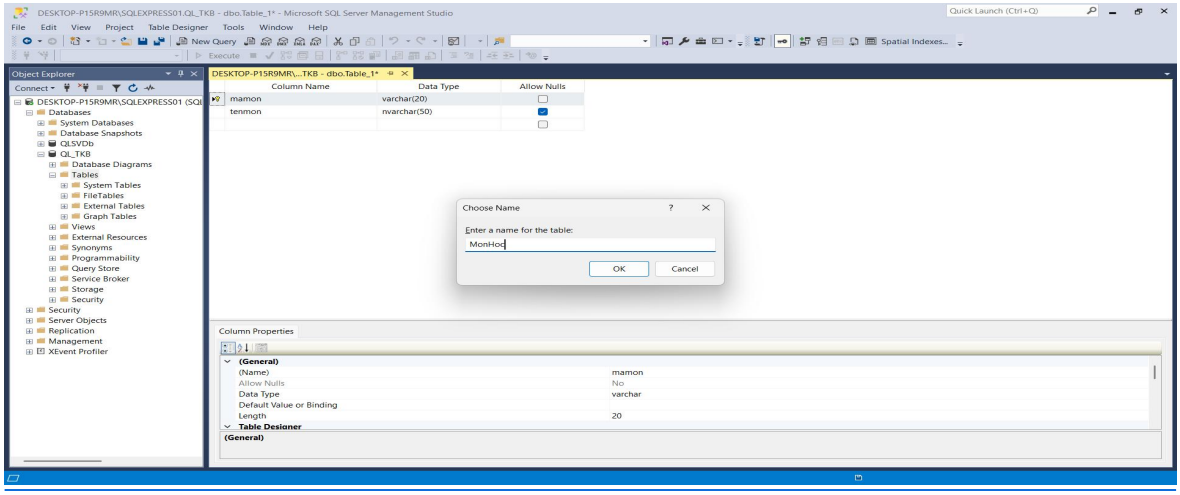
4.2 CÁC BƯỚC THỰC HIỆN

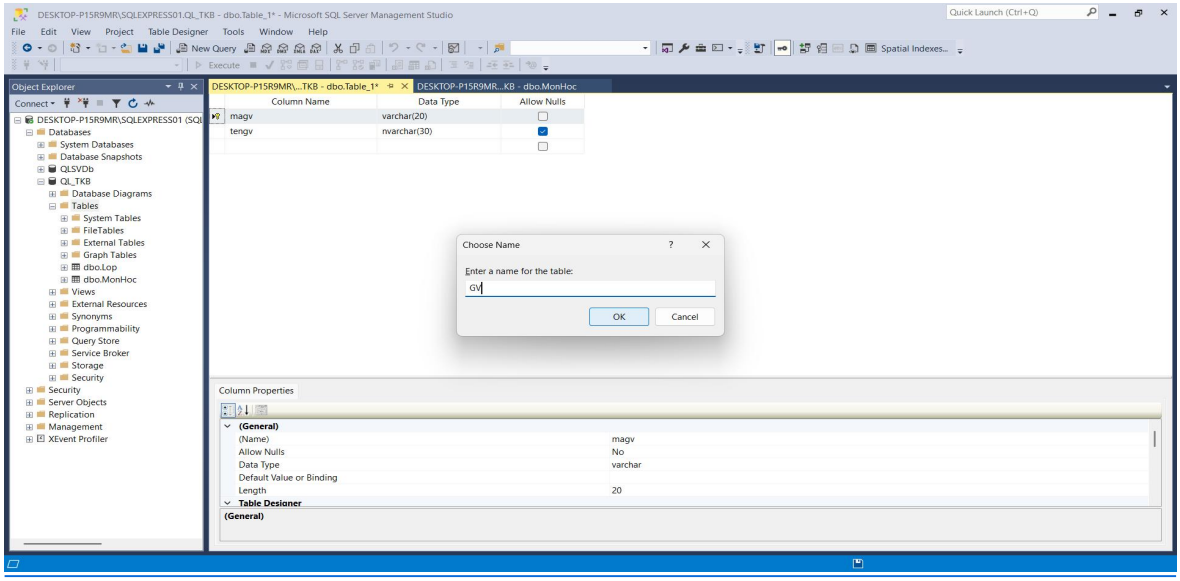
Bước 1: Tạo Database mới có tên là QL_TKB từng bước như các bài tập trước



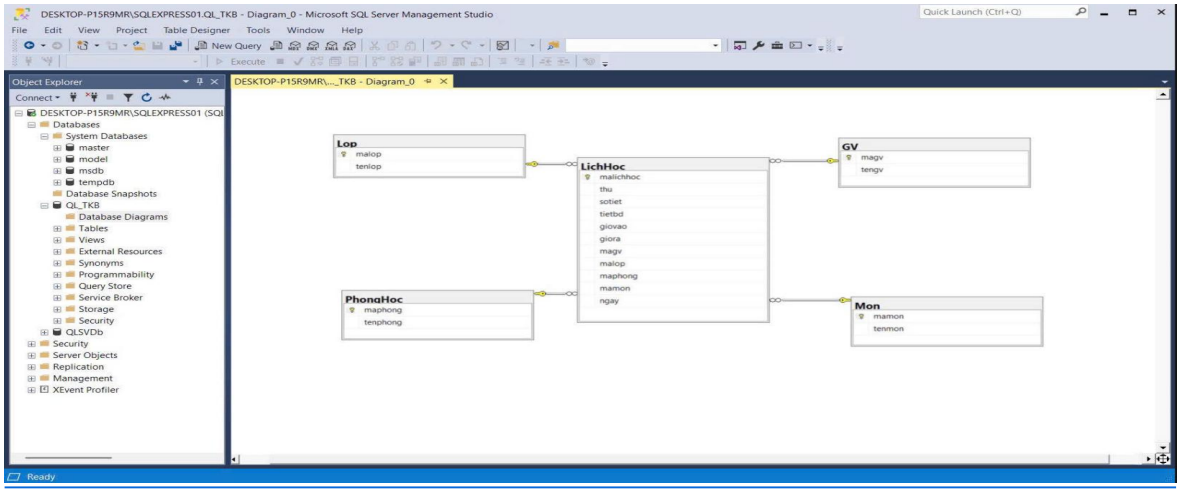
Bước 2: Tạo các bảng có tên: MonHoc, Lop, GV, PhongHoc, LichHoc theo chuẩn 3NF:







Bước 3: Thực hiện đặt các khóa chinh và các khóa ngoại liên kết để thu được sơ đồ liên kết:



Bước 4: Copy dữ liệu từ nguồn TMS.tnut.edu.vn, patse vào Excel sau đó thực hiện dán các dữ liệu cần thiết vào các bảng ở mục edit bảng trong sql:

maphong	tenphong
1	A9-201
10	A10-201
11	A8-101
12	A10-102
13	A10-503
14	A9-102
15	A9-302
16	A16-501
17	A10-502
18	A10-202
19	A7-203
2	A9-204
20	A9-301
21	A7-204
22	A16-301
23	TVDT-T203
24	TVDT-T202
25	TVDT-T301
3	A10-101
4	A10-104
5	A16-603
6	A10-401
7	A8-303
8	A16-401
9	A10-404
NULL	NULL

magv	tengv
35	Đ.T.Hiên
36	N.V.Huy
37	N.T.Hương
38	T.T.N.Linh
39	P.T.T.Hiền
40	T.C.Nhung
41	N.T.Linh
42	N.V.Tính
43	T.T.Thanh
44	Đ.D.Cốp
45	N.M.Ngọc
46	B.T.Thi
47	P.Q.Hiếu
48	Đ.N.Tôn
49	L.T.H.Trang
50	N.T.Duy
NULL	NULL

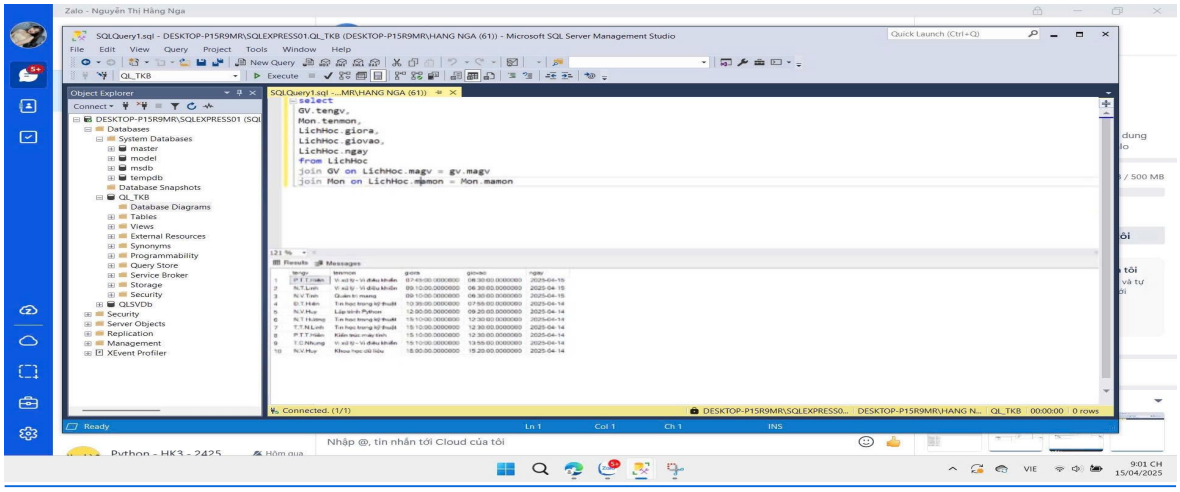
	mamon	tenmon
▶	1	Tin học tron...
	10	Công nghệ ...
	11	Phương phá...
	12	Toán rời rạc
	2	Lập trình Py...
	3	Kiến trúc m...
	4	Vi xử lý – Vi ...
	5	Khoa học d...
	6	Quản trị mạ...
	7	Chuyển đổi ...
	8	Phân tích và...
	9	Hệ quản trị ...
*	NULL	NULL

	malop	tenlop
▶	1	58KTP
	10	60CNTDH5
	11	60CNTDH1
	12	59KC2
	13	59KC1
	14	58CDT2
	15	60CNTDH7
	16	60CNTDH3
	17	60KMT
	18	58CLCDT
	19	60CNTDH4
	2	60CNTDH2
	20	59KXD
	21	59KC1-02
	22	58KTD1-02
	23	58CDT2-01
	24	59KC2-01
	25	58CDT3-03
	26	58CDT3-02
	27	59KC2-02
	28	59KMT-01
	29	60KMT-01
	3	59KMT
	30	60CNTDH3-...
	31	58CDT3-01
	32	60KMT-03
	33	60CNTDH4-...
	34	60CNTDH6-...
	35	58CDT1-01

malop	tenlop
40	60CNTDH4-...
41	60CNTDH6-...
42	58CDT1-02
43	60CNTDH5-...
44	59KC1-01
45	58KTD2-02
46	57KMT-01
47	60CNTDH2-...
48	59KMT-02
49	60CNTDH7-...
5	58CDT1
50	58KTD1-01
51	60CNTDH1-...
52	60CNTDH2-...
53	59KMT-03
54	60CNTDH7-...
55	58CDT2-02
56	58KTP-01
57	60CNTDH1-...
58	58CLCDT-01
59	60KMT-02
6	57KMT
60	60CNTDH1-...
61	58KTP-03
62	58KTP-02
7	58CDT3
8	58KTD2
9	58KTD1
NULL	NULL

malichhoc	thu	sotiet	tietbd	giovao	giora	magv	malop	maphong	mamon	ngay
10	3	1	1	06:30:00	07:45:00	39	8	8	4	2025-04-15
11	3	2	1	06:30:00	09:10:00	41	9	9	4	2025-04-15
12	3	2	1	06:30:00	09:10:00	42	2	2	6	2025-04-15
3	2	2	2	07:55:00	10:35:00	35	1	1	1	2025-04-14
4	2	2	3	09:20:00	12:00:00	36	2	2	2	2025-04-14
5	2	2	5	12:30:00	15:10:00	37	3	3	1	2025-04-14
6	2	2	5	12:30:00	15:10:00	38	4	4	1	2025-04-14
7	2	2	5	12:30:00	15:10:00	39	5	5	3	2025-04-14
8	2	1	6	13:55:00	15:10:00	40	6	6	4	2025-04-14
9	2	2	7	15:20:00	18:00:00	36	7	7	5	2025-04-14
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Bước 5: 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:

4.3 LINK TRA CỨU

1 Link github:

https://github.com/Nganguyen211/baitap4/blob/main/README.md

2 Mã qr :

CHƯƠNG 5 TRINGGER

5.1 YÊU CẦU

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.

5.2 CÁC BƯỚC THỰC HIỆN

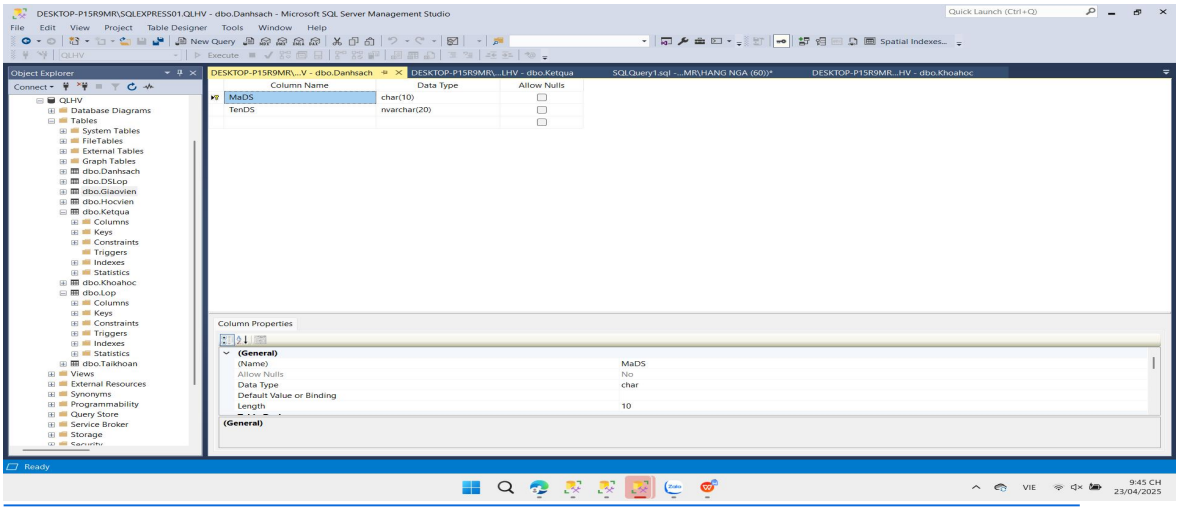
Đề bài đồ án : Quản lý học viên tại trung tâm ngoại ngữ ETC yêu cầu của bài toán:

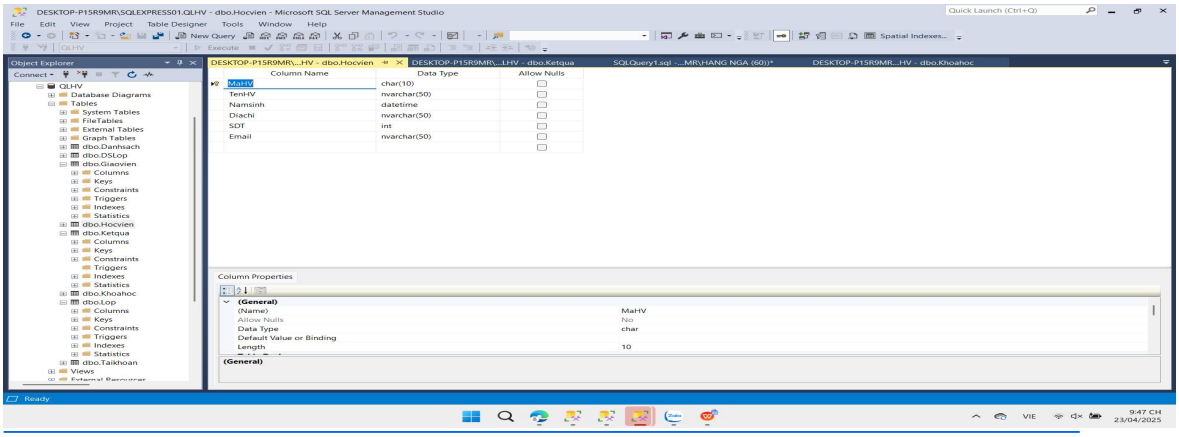
- Khảo sát, phân tích hiện trạng của hệ thông quản lý học viên tại trung tâm ngoại ngữ etc

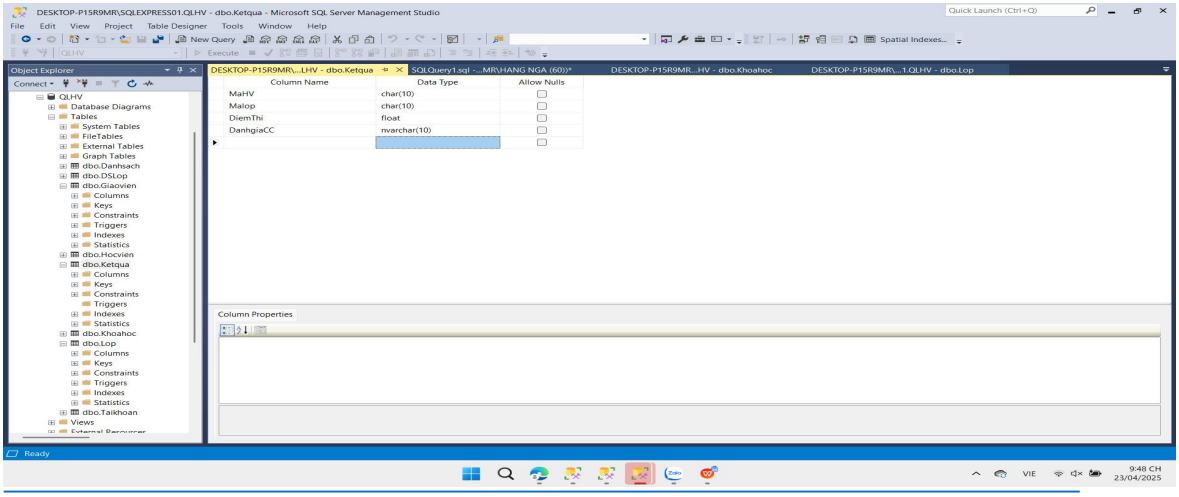
- Phân tích hệ thống thông tin quản lý học viên tại trung tâm ngoại ngữ etc

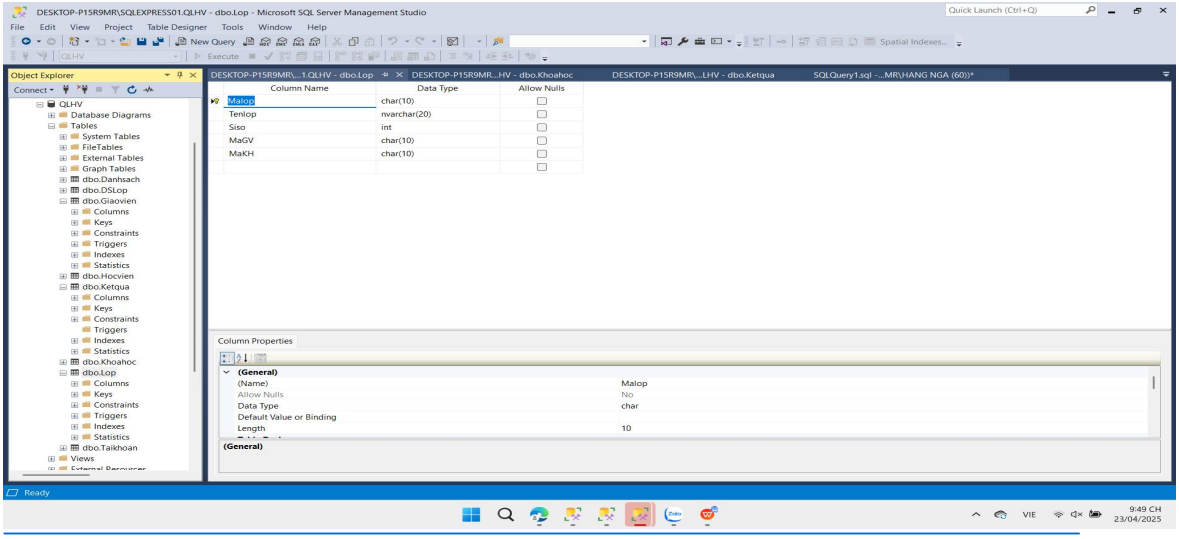
- Phân tích hệ thống quản lý học viên tại trung tâm ngoại ngữ etc Xây dựng các biểu đồ usecase ,tuần tự,trạng thái ,hoạt động -Xây dựng các tệp cơ sở dữ liệu.

- Thiết kế chương trình Trước hết cần tạo database QLHV trong đó bao gồm các bảng

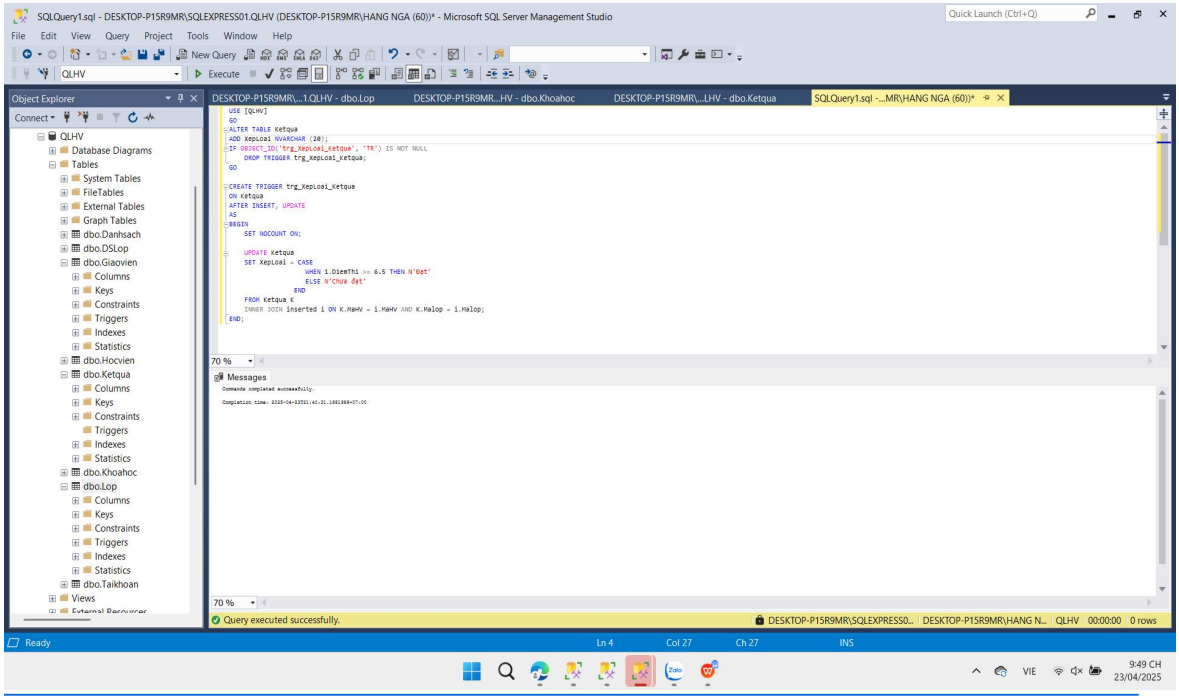




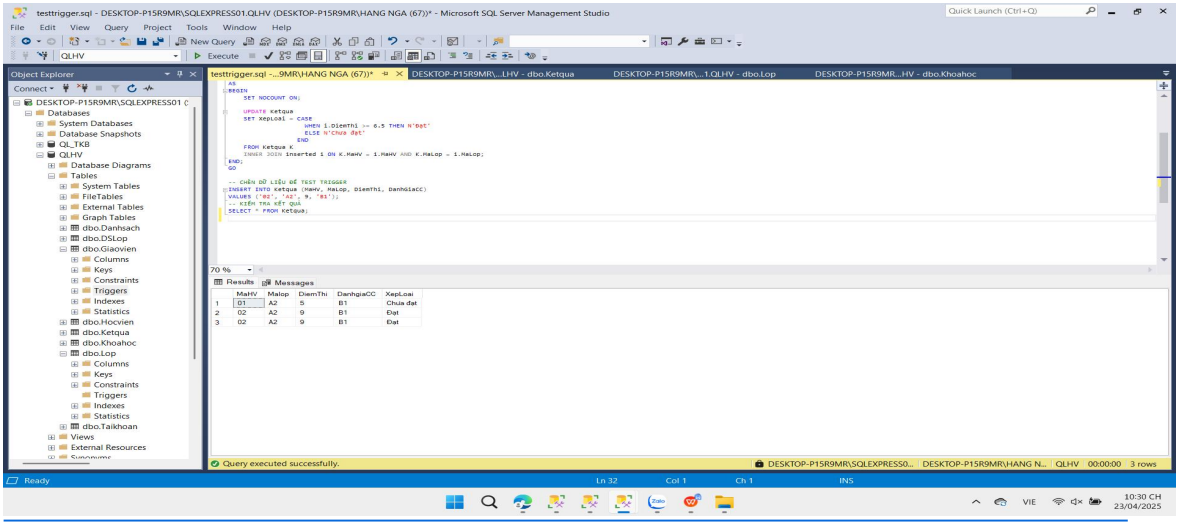




Tiếp theo là thêm trường phi chuẩn XepLoai vào bảng Ketqua và viết tringger test bảng này với trường XepLoai mục tiêu của việc này là phân loại được học viên có đạt chứng chỉ b1 hay không và đang ở trình độ nào nếu Diemthi>=6.5 thì XepLoai sẽ là Đạt và ngược lại



Nhập dữ liệu và test thử



Trong đồ án quản lý học viên tại trung tâm ngoại ngữ, trigger có thể được sử dụng để tự động hóa nhiều tác vụ quan trọng, ví dụ:

- Tự động cập nhật thông tin liên quan khi có sự thay đổi:

  - Khi một học viên đăng ký một khóa học mới, trigger có thể tự động cập nhật số lượng học viên trong khóa học đó.

  - Khi thông tin cá nhân của học viên (ví dụ: số điện thoại, địa chỉ) được cập nhật, trigger có thể tự động cập nhật thông tin này ở tất cả các bảng liên quan

- Đảm bảo tính toàn vẹn dữ liệu:

  - Khi một học viên bị xóa khỏi hệ thống, trigger có thể tự động xóa tất cả các bản ghi liên quan đến học viên đó trong các bảng khác

  - Trigger có thể được sử dụng để kiểm tra các ràng buộc phức tạp hơn mà các ràng buộc thông thường (ví dụ: NOT NULL, UNIQUE) không thể xử lý. Ví dụ, đảm bảo rằng một học viên không thể đăng ký hai khóa học trùng lịch.

- Tự động ghi log (nhật ký) các hoạt động quan trọng:

  - Khi có một học viên mới được thêm vào, một khóa học được tạo, hoặc một khoản thanh toán được thực hiện, trigger có thể tự động ghi lại thời gian, người thực hiện và chi tiết của hành động này vào một bảng nhật ký. Điều này giúp theo dõi lịch sử hoạt động của hệ thống và hỗ trợ việc kiểm tra, bảo trì sau này.

- Tự động thực hiện các nghiệp vụ khác:

  - Khi học viên hoàn thành một khóa học, trigger có thể tự động cập nhật trạng thái học viên và có thể gửi thông báo (ví dụ: email) về việc hoàn thành khóa học.

*Mã code cấu trúc sql

```
USE [master]
GO
/****** Object:  Database [QLHV]    Script Date: 23/04/2025 10:32:12 CH ******/
CREATE DATABASE [QLHV]
 CONTAINMENT = NONE
 ON  PRIMARY
( NAME = N'QLHV', FILENAME = N'D:\DOWNLOAD\MSSQL16.SQLEXPRESS01\MSSQL\DATA\QLHV.mdf' , SIZE = 8192KB , MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )
 LOG ON
( NAME = N'QLHV_log', FILENAME = N'D:\DOWNLOAD\MSSQL16.SQLEXPRESS01\MSSQL\DATA\QLHV_log.ldf' , SIZE = 8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB )
 WITH CATALOG_COLLATION = DATABASE_DEFAULT, LEDGER = OFF
GO
ALTER DATABASE [QLHV] SET COMPATIBILITY_LEVEL = 160
GO
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [QLHV].[dbo].[sp_fulltext_database] @action = 'enable'
```

```
end
GO
ALTER DATABASE [QLHV] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [QLHV] SET ANSI_NULLS OFF
GO
ALTER DATABASE [QLHV] SET ANSI_PADDING OFF
GO
ALTER DATABASE [QLHV] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [QLHV] SET ARITHABORT OFF
GO
ALTER DATABASE [QLHV] SET AUTO_CLOSE OFF
GO
ALTER DATABASE [QLHV] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [QLHV] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [QLHV] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [QLHV] SET CURSOR_DEFAULT  GLOBAL
GO
ALTER DATABASE [QLHV] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [QLHV] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [QLHV] SET QUOTED_IDENTIFIER OFF
GO
ALTER DATABASE [QLHV] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [QLHV] SET  DISABLE_BROKER
GO
ALTER DATABASE [QLHV] SET AUTO_UPDATE_STATISTICS_ASYNC OFF
GO
ALTER DATABASE [QLHV] SET DATE_CORRELATION_OPTIMIZATION OFF
GO
ALTER DATABASE [QLHV] SET TRUSTWORTHY OFF
```

```
GO
ALTER DATABASE [QLHV] SET ALLOW_SNAPSHOT_ISOLATION OFF
GO
ALTER DATABASE [QLHV] SET PARAMETERIZATION SIMPLE
GO
ALTER DATABASE [QLHV] SET READ_COMMITTED_SNAPSHOT OFF
GO
ALTER DATABASE [QLHV] SET HONOR_BROKER_PRIORITY OFF
GO
ALTER DATABASE [QLHV] SET RECOVERY SIMPLE
GO
ALTER DATABASE [QLHV] SET  MULTI_USER
GO
ALTER DATABASE [QLHV] SET PAGE_VERIFY CHECKSUM
GO
ALTER DATABASE [QLHV] SET DB_CHAINING OFF
GO
ALTER DATABASE [QLHV] SET FILESTREAM( NON_TRANSACTED_ACCESS = OFF )
GO
ALTER DATABASE [QLHV] SET TARGET_RECOVERY_TIME = 60 SECONDS
GO
ALTER DATABASE [QLHV] SET DELAYED_DURABILITY = DISABLED
GO
ALTER DATABASE [QLHV] SET ACCELERATED_DATABASE_RECOVERY = OFF
GO
ALTER DATABASE [QLHV] SET QUERY_STORE = ON
GO
ALTER DATABASE [QLHV] SET QUERY_STORE (OPERATION_MODE = READ_WRITE,
CLEANUP_POLICY = (STALE_QUERY_THRESHOLD_DAYS = 30),
DATA_FLUSH_INTERVAL_SECONDS = 900, INTERVAL_LENGTH_MINUTES = 60,
MAX_STORAGE_SIZE_MB = 1000, QUERY_CAPTURE_MODE = AUTO,
SIZE_BASED_CLEANUP_MODE = AUTO, MAX_PLANS_PER_QUERY = 200,
WAIT_STATS_CAPTURE_MODE = ON)
GO
USE [QLHV]
GO
/****** Object:  Table [dbo].[Danhsach]    Script Date: 23/04/2025 10:32:13 CH ******/
```

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Danhsach](
	[MaDS] [char](10) NOT NULL,
	[TenDS] [nvarchar](20) NOT NULL,
 CONSTRAINT [PK_Danhsach] PRIMARY KEY CLUSTERED
(
	[MaDS] 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
/****** Object:  Table [dbo].[DSLop]    Script Date: 23/04/2025 10:32:13 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[DSLop](
	[MaDS] [char](10) NOT NULL,
	[Malop] [char](10) NOT NULL
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[Giaovien]    Script Date: 23/04/2025 10:32:13 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Giaovien](
	[MaGV] [char](10) NOT NULL,
	[TenGV] [nvarchar](50) NOT NULL,
	[SDT] [char](20) NOT NULL,
	[Trinhdo] [nvarchar](20) NOT NULL,
 CONSTRAINT [PK_Giaovien] PRIMARY KEY CLUSTERED
(
```

```
	[MaGV] 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
/****** Object:  Table [dbo].[Hocvien]    Script Date: 23/04/2025 10:32:13 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Hocvien](
	[MaHV] [char](10) NOT NULL,
	[TenHV] [nvarchar](50) NOT NULL,
	[Namsinh] [datetime] NOT NULL,
	[Diachi] [nvarchar](50) NOT NULL,
	[SDT] [int] NOT NULL,
	[Email] [nvarchar](50) NOT NULL,
 CONSTRAINT [PK_Hocvien] PRIMARY KEY CLUSTERED
(
	[MaHV] 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
/****** Object:  Table [dbo].[Ketqua]    Script Date: 23/04/2025 10:32:13 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Ketqua](
	[MaHV] [char](10) NOT NULL,
	[Malop] [char](10) NOT NULL,
	[DiemThi] [float] NOT NULL,
	[DanhgiaCC] [nvarchar](10) NOT NULL,
	[XepLoai] [nvarchar](20) NULL
```

```
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[Khoahoc]    Script Date: 23/04/2025 10:32:13 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Khoahoc](
	[MaKH] [char](10) NOT NULL,
	[TenKH] [nvarchar](20) NOT NULL,
	[Ngaybatdau] [date] NOT NULL,
	[Ngayketthuc] [date] NOT NULL,
	[Hocphi] [money] NOT NULL,
 CONSTRAINT [PK_Khoahoc] PRIMARY KEY CLUSTERED
(
	[MaKH] 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
/****** Object:  Table [dbo].[Lop]    Script Date: 23/04/2025 10:32:13 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Lop](
	[Malop] [char](10) NOT NULL,
	[Tenlop] [nvarchar](20) NOT NULL,
	[Siso] [int] NOT NULL,
	[MaGV] [char](10) NOT NULL,
	[MaKH] [char](10) NOT NULL,
 CONSTRAINT [PK_Lop] PRIMARY KEY CLUSTERED
(
	[Malop] 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
/****** Object:  Table [dbo].[Taikhoan]    Script Date: 23/04/2025 10:32:13 CH ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Taikhoan](
	[MaTk] [char](10) NOT NULL,
	[Matkhau] [char](10) NOT NULL,
	[TrangthaiTK] [nvarchar](50) NOT NULL,
	[MaHV] [char](10) NOT NULL,
 CONSTRAINT [PK_Taikhoan] PRIMARY KEY CLUSTERED
(
	[MaTk] 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].[DSLop]  WITH CHECK ADD  CONSTRAINT [FK_DSLop_Danhsach] FOREIGN KEY([MaDS])
REFERENCES [dbo].[Danhsach] ([MaDS])
GO
ALTER TABLE [dbo].[DSLop] CHECK CONSTRAINT [FK_DSLop_Danhsach]
GO
ALTER TABLE [dbo].[DSLop]  WITH CHECK ADD  CONSTRAINT [FK_DSLop_Lop] FOREIGN KEY([Malop])
REFERENCES [dbo].[Lop] ([Malop])
GO
ALTER TABLE [dbo].[DSLop] CHECK CONSTRAINT [FK_DSLop_Lop]
GO
ALTER TABLE [dbo].[Ketqua]  WITH CHECK ADD  CONSTRAINT [FK_Ketqua_Hocvien] FOREIGN KEY([MaHV])
```

```
REFERENCES [dbo].[Hocvien] ([MaHV])
GO
ALTER TABLE [dbo].[Ketqua] CHECK CONSTRAINT [FK_Ketqua_Hocvien]
GO
ALTER TABLE [dbo].[Ketqua] WITH CHECK ADD CONSTRAINT [FK_Ketqua_Lop] FOREIGN KEY([Malop])
REFERENCES [dbo].[Lop] ([Malop])
GO
ALTER TABLE [dbo].[Ketqua] CHECK CONSTRAINT [FK_Ketqua_Lop]
GO
ALTER TABLE [dbo].[Lop] WITH CHECK ADD CONSTRAINT [FK_Lop_Giaovien] FOREIGN KEY([MaGV])
REFERENCES [dbo].[Giaovien] ([MaGV])
GO
ALTER TABLE [dbo].[Lop] CHECK CONSTRAINT [FK_Lop_Giaovien]
GO
ALTER TABLE [dbo].[Lop] WITH CHECK ADD CONSTRAINT [FK_Lop_Khoahoc] FOREIGN KEY([MaKH])
REFERENCES [dbo].[Khoahoc] ([MaKH])
GO
ALTER TABLE [dbo].[Lop] CHECK CONSTRAINT [FK_Lop_Khoahoc]
GO
ALTER TABLE [dbo].[Lop] WITH CHECK ADD CONSTRAINT [FK_Lop_Lop] FOREIGN KEY([Malop])
REFERENCES [dbo].[Lop] ([Malop])
GO
ALTER TABLE [dbo].[Lop] CHECK CONSTRAINT [FK_Lop_Lop]
GO
ALTER TABLE [dbo].[Taikhoan] WITH CHECK ADD CONSTRAINT [FK_Taikhoan_Hocvien] FOREIGN KEY([MaHV])
REFERENCES [dbo].[Hocvien] ([MaHV])
GO
ALTER TABLE [dbo].[Taikhoan] CHECK CONSTRAINT [FK_Taikhoan_Hocvien]
GO
USE [master]
GO
ALTER DATABASE [QLHV] SET READ_WRITE
```

```
GO
```

*Mã data tringger sql

```
USE [QLHV]
GO

-- XÓA TRIGGER NẾU ĐÃ CÓ
IF OBJECT_ID('trg_XepLoai_Ketqua', 'TR') IS NOT NULL
   DROP TRIGGER trg_XepLoai_Ketqua;
GO

-- TẠO TRIGGER
CREATE TRIGGER trg_XepLoai_Ketqua
ON Ketqua
AFTER INSERT, UPDATE
AS
BEGIN
   SET NOCOUNT ON;

   UPDATE Ketqua
   SET XepLoai = CASE
                 WHEN i.DiemThi >= 6.5 THEN N'Đạt'
                 ELSE N'Chưa đạt'
                END
   FROM Ketqua K
   INNER JOIN inserted i ON K.MaHV = i.MaHV AND K.MaLop = i.MaLop;
END;
GO
-- CHÈN DỮ LIỆU ĐỂ TEST TRIGGER
INSERT INTO Ketqua (MaHV, MaLop, DiemThi, DanhGiaCC)
VALUES ('02', 'A2', 9, 'B1');
-- KIỂM TRA KẾT QUẢ
SELECT * FROM Ketqua;
```

5.3 LINK TRA CỨU

1 Link github:

baitap5/README.md at main · Nganguyen211/baitap5

2 Mã qr :

CHƯƠNG 6 TRUY VẤN DỮ LIỆU

6.1 YÊU CẦU

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. Viết câu lệnh sql để liệt kê tất cả sinh viên nữ thuộc ngành kmt có trong bảng sinhvien.

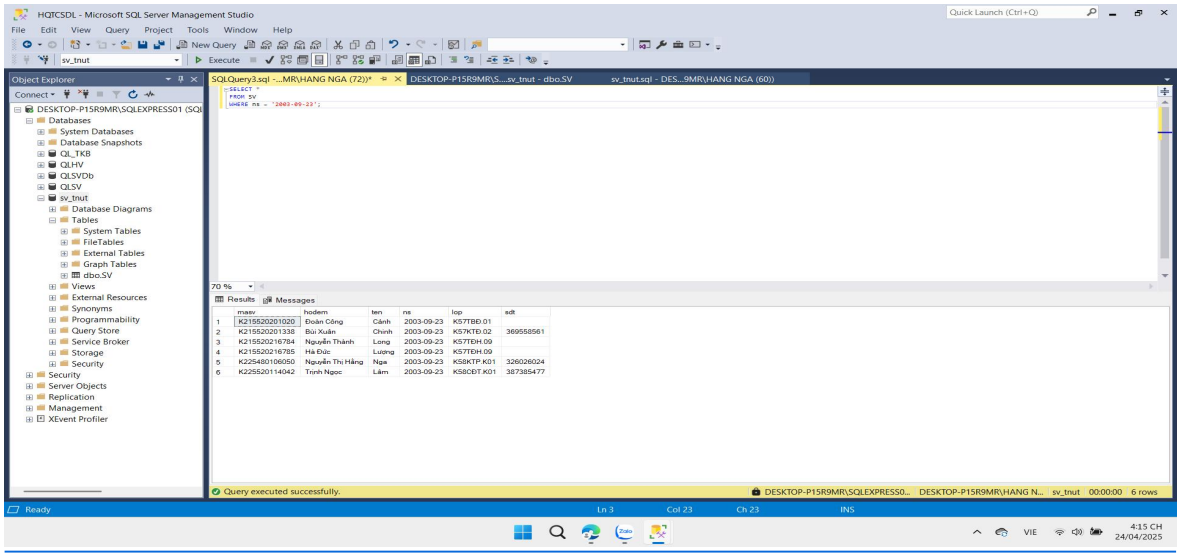
6.2 CÁC BƯỚC THỰC HIỆN

BÀI LÀM

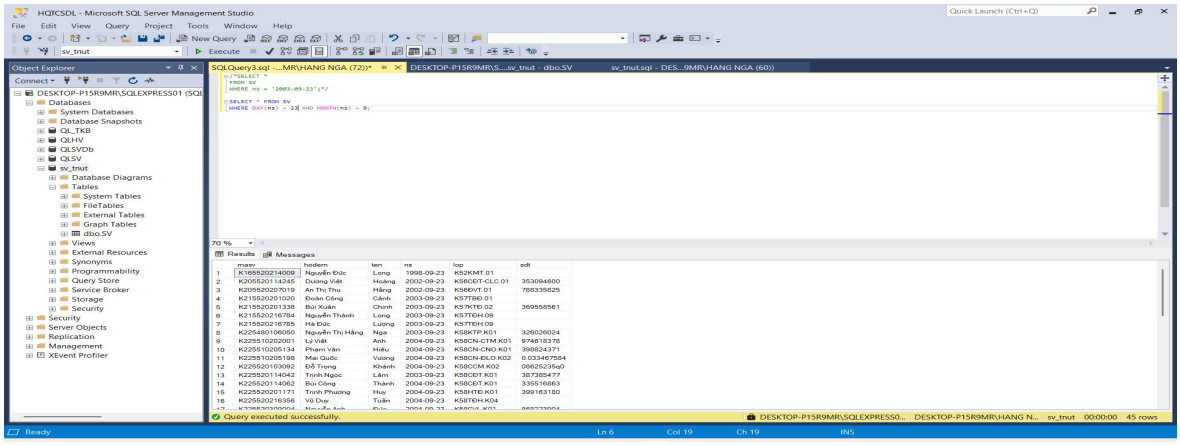
1 Tạo cơ sở sv_tnut dể mở file sv_tnut.sql tiếp theo chọn File > Open > File > Chọn sv_tnut.sql. Chạy toàn bộ script: Nhấn F5 hoặc nút "Execute".

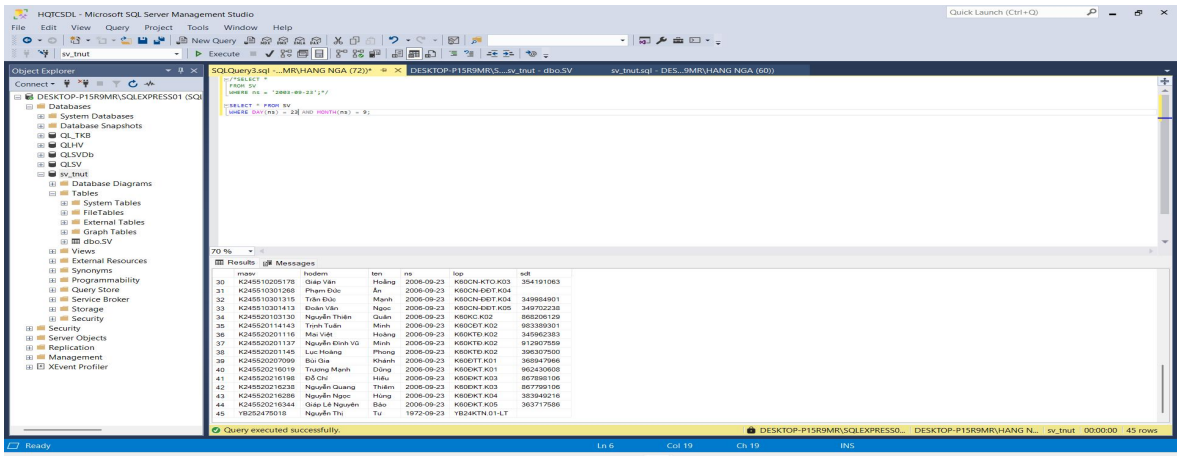
2 Dữ liệu đầu bài của bản thân Nguyễn Thị Hằng Nga sdt 0326026024 ngày tháng năm sinh 23/09/2003

3 Nhập sql để tìm xem có những sv nào trùng hoàn toàn ngày 23/09/2003



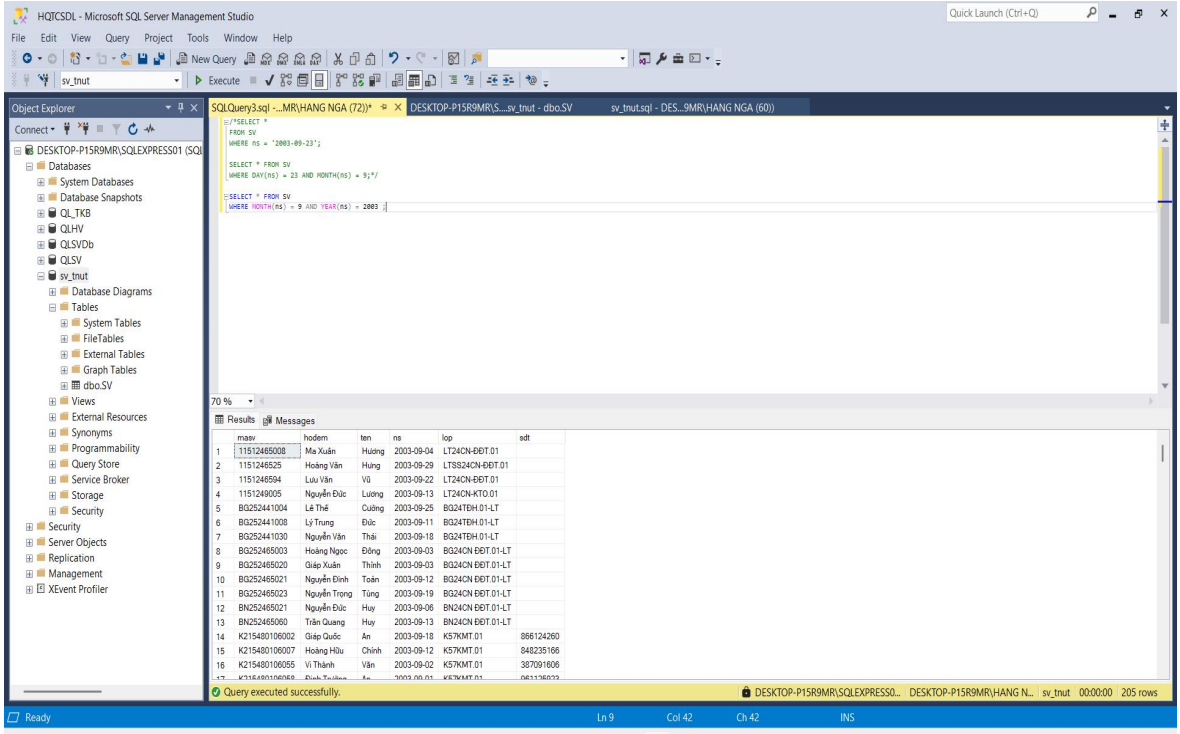
4 Nhập sql để tìm xem có những sv nào trùng ngày và tháng sinh 23/09

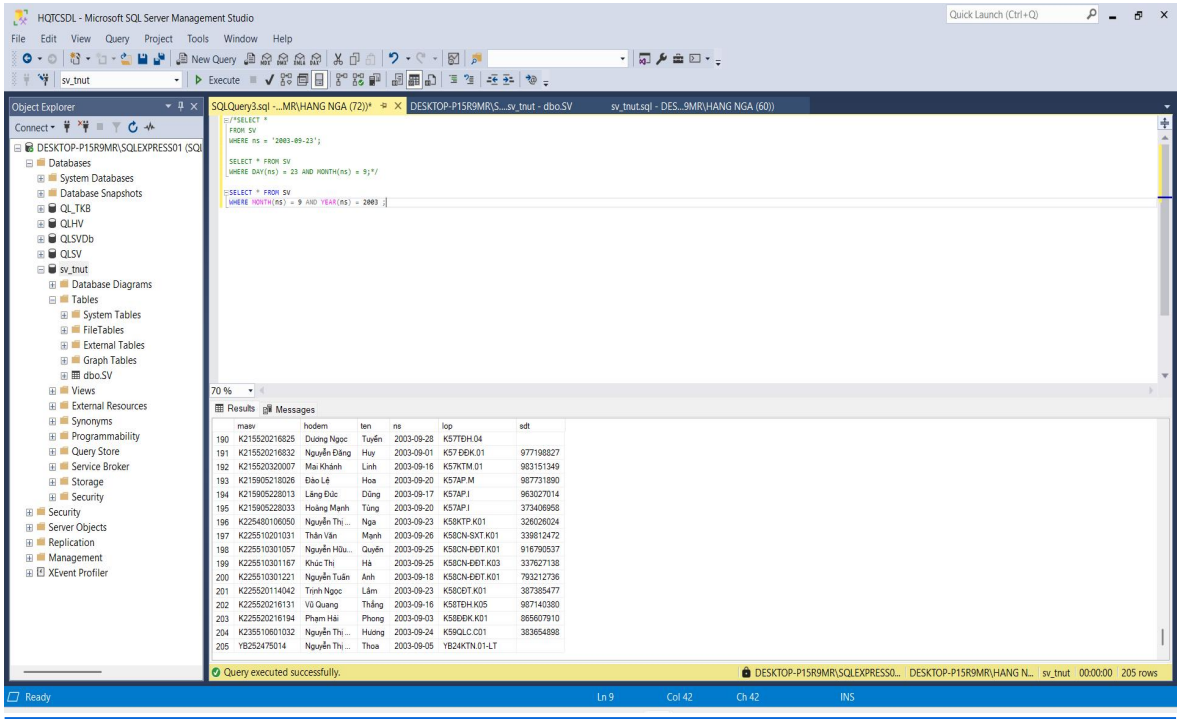




bao gồm 45 sv có cùng ngày tháng với em

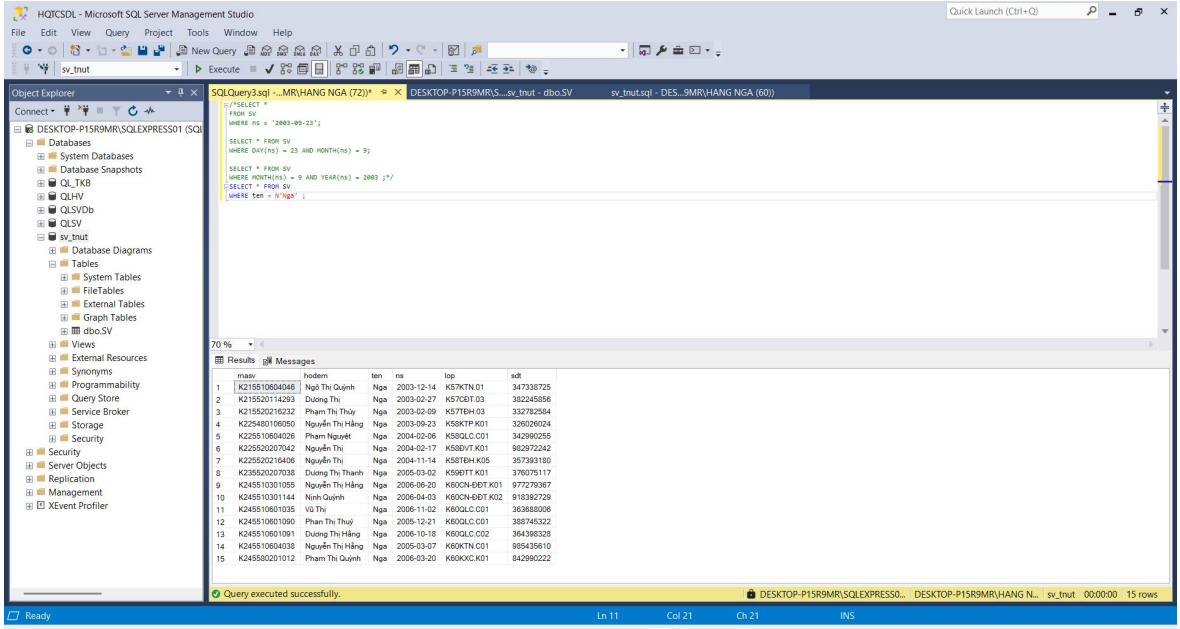
5 Nhập sql để tìm xem có những sv nào trùng tháng và năm sinh



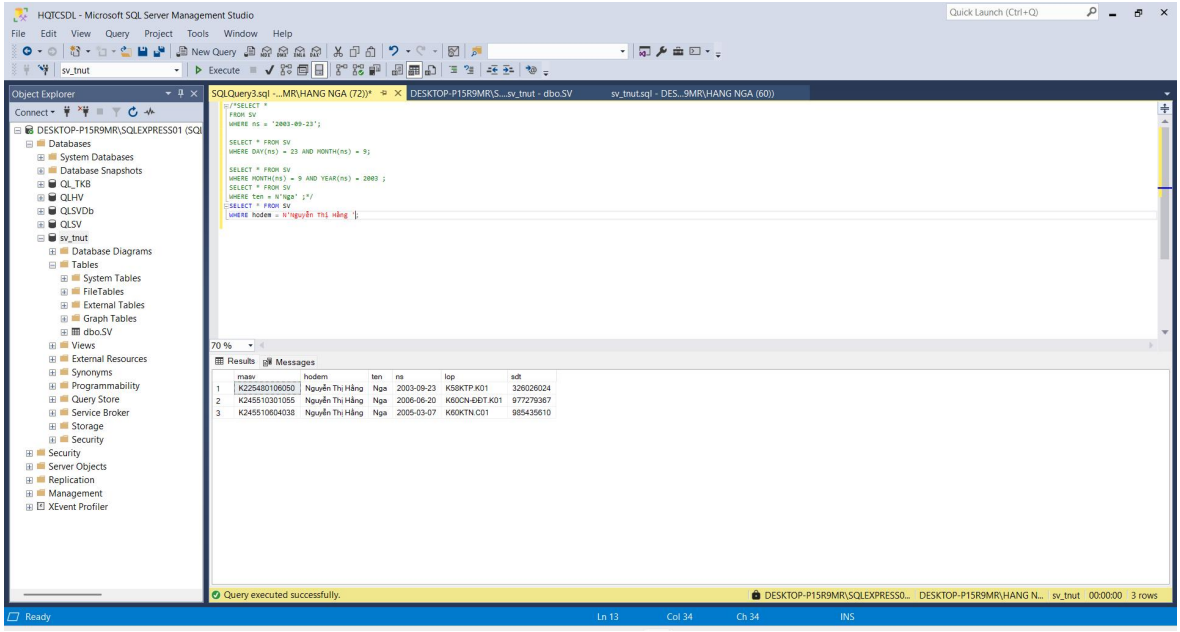


bao gồm 205 bạn sv có cùng tháng và năm sinh với em

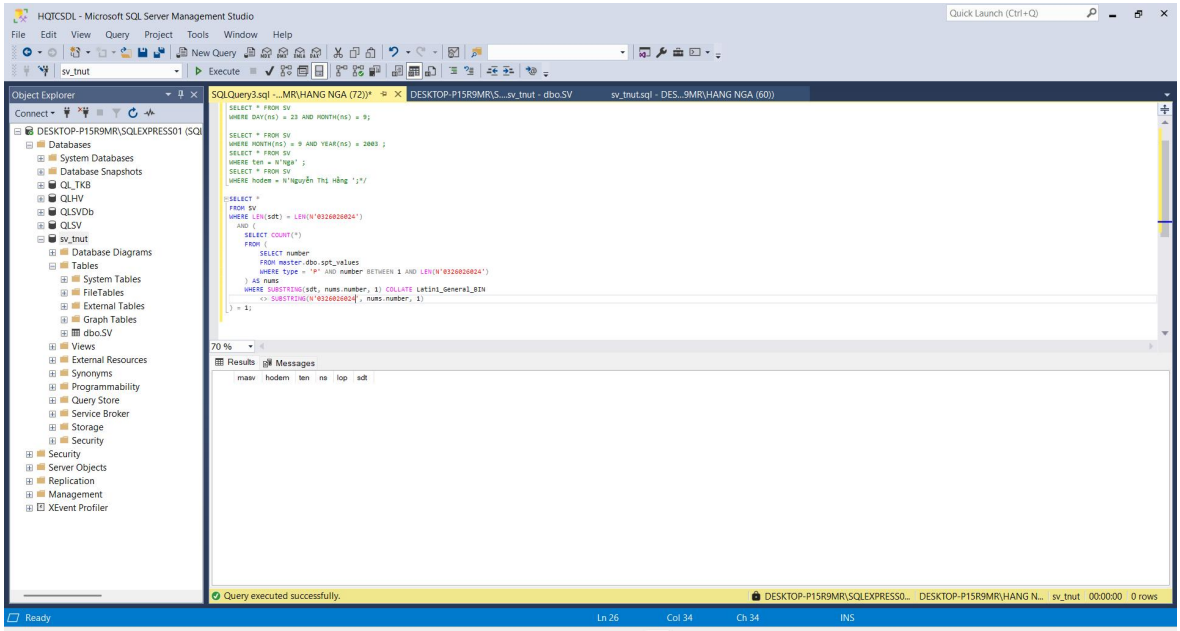
6 Nhập sql để tìm xem có những sv nào trùng tên



7 Nhập sql để tìm xem có những sv nào trùng họ và tên đệm

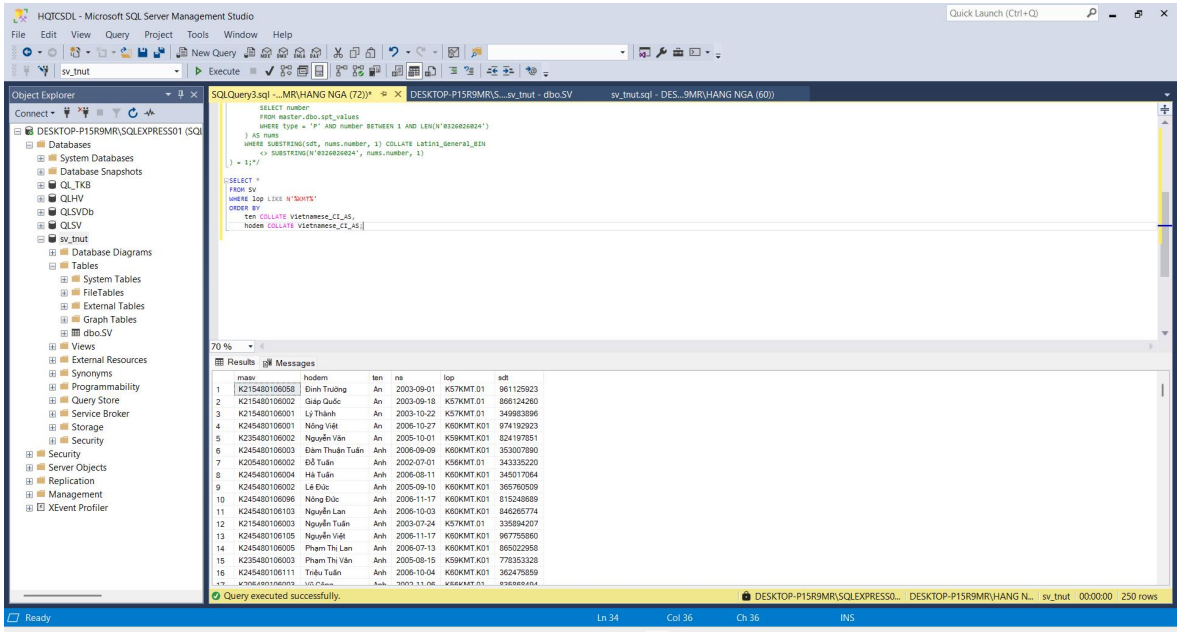


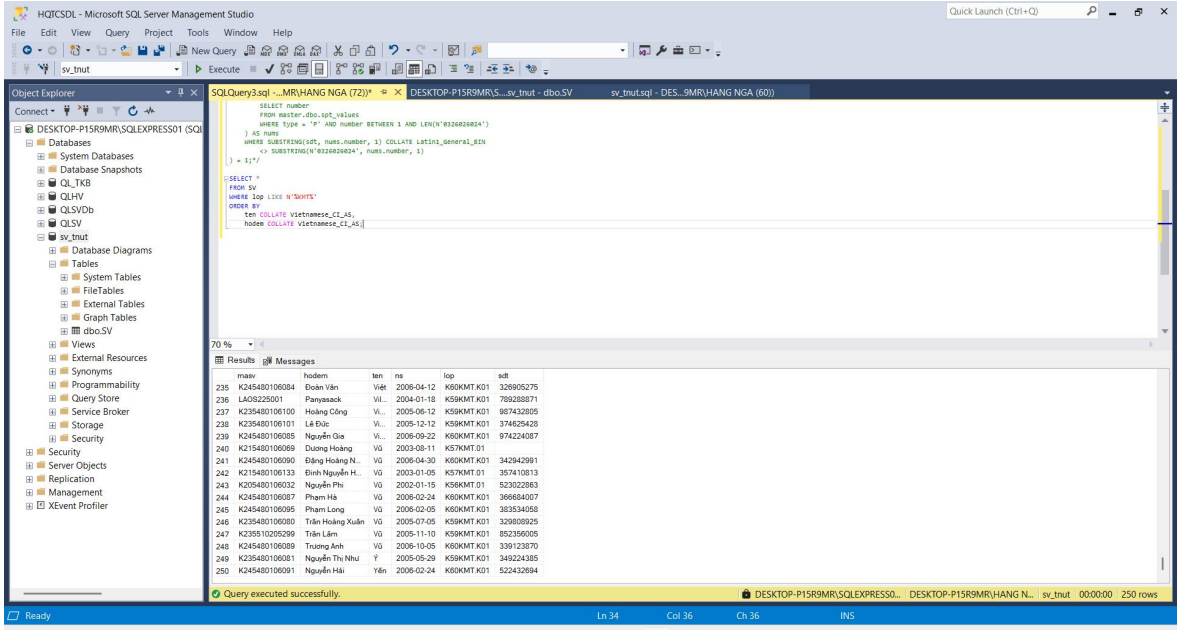
8 Nhập sql để tìm xem có những sv nào có sđt sai khác chỉ 1 số so với sđt của em.



không có ai khác số đt em 1 số

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



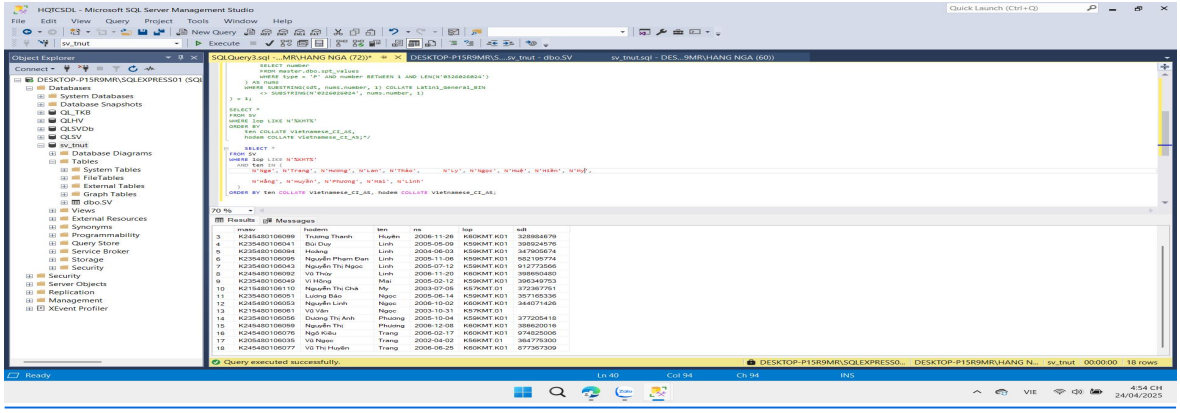


gồm 250 sv

* giải thích : lop LIKE N'%KMT%': lọc các sinh viên thuộc ngành KMT. Có thể bạn dùng tên lớp chứa từ "KMT". ORDER BY ten, hodem: sắp xếp trước theo tên, sau đó theo họ đệm. COLLATE Vietnamese_CI_AS: ép SQL Server sắp xếp theo bảng mã tiếng Việt chuẩn (phân biệt đúng thứ tự chữ có dấu như: a, á, à, ă, â, b, c,...)

10 Viết câu lệnh sql để liệt kê tất cả sinh viên nữ thuộc ngành kmt có trong bảng sinhvien.

Vì trong bảng không có trường giới tính nên chỉ có thể phỏng đoán theo một số tên phổ biến là nữ như Linh ,Nga ,Lan, Hương,Huệ ,Huyền,My ,Ly,Ngọc,Trang,Phương ,Hằng, Thảo,Mai, Lan,Hiền nếu có thêm trường giới tính thì sẽ dễ truy vấn hơn



6.3 LINK TRA CỨU

1 Link github:

Nganguyen211/battap6: Hệ cơ sở quản trị dữ liệu - Nga58KTP_K225480106050

2 Mã qr :