

TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP  
KHOA ĐIỆN TỬ



**BÀI TẬP LỚN MÔN HỆ QUẢN TRỊ CSDL**

**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 MÔN HỆ QUẢN TRỊ CSDL**  
**BỘ MÔN : CÔNG NGHỆ THÔNG TIN**

**GIÁO VIÊN HƯỚNG DẪN** : Th Đỗ Duy Cốp

**HỌ VÀ TÊN SINH VIÊN** : Nguyễn Giang Long

**LỚP** : K58KTP

**MSSV** : K225480106043

**THÁI NGUYÊN - 2025**

---

**TRƯỜNG ĐHKT&CN CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  
**KHOA ĐIỆN TỬ** **Độc lập - Tự do - Hạnh phúc**

**NHIỆM VỤ BÀI TẬP LỚN**

**Sinh viên:** Nguyễn Giang Long

**Lớp:** K58KMT

**Khoá:** 2024-2025

**Bộ môn:** Hệ quản trị CSDL

**Giáo viên hướng dẫn:** Thầy Đỗ Duy Cốp

1. Tên: bài tập lớn Hệ quản trị CSDL
2. Trình bày các bài tập đã được giao
3. Ngày hoàn thành nhiệm vụ: 07/06/2025

**BCN KHOA**

(Ký và ghi rõ họ tên)

**TRƯỜNG BỘ MÔN**

(Ký và ghi rõ họ tên)

**GIÁO VIÊN HƯỚNG DẪN**

(Ký và ghi rõ họ tên)

**PHIẾU GHI ĐIỂM  
HƯỚNG DẪN BÀI TẬP LỚN**

**Sinh viên:** Nguyễn Giang Long

**Lớp:** K58KMT

**GVHD:** 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

Bài tập về nhà 01 - môn Hệ quản trị csdl:.....	4
BÀI TẬP VỀ NHÀ 02 - MÔN HỆ QUẢN TRỊ CSDL: .....	5
BÀI TẬP VỀ NHÀ 03 - MÔN HỆ QUẢN TRỊ CSDL: .....	13
Bài tập 4 : (sql server) .....	30
BÀI TẬP VỀ NHÀ 05, Môn Hệ quản trị csdl. .....	38
Bài tập 6: Hệ quản trị CSDL .....	48

---

### **Bài tập về nhà 01 - môn Hệ quản trị csdl:**

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 tùy 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 management studio (bản mới nhất)
3. Các bước để đăng nhập từ sql management 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 (để, 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 (gọi 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)

Hình thức làm bài:

- 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
- Mỗi hình nên paste vào PowerPoint, có thể dùng ai hỗ trợ (chatgpt , gemini, copilot, ...) để hiểu hết ý nghĩa các phần trên ảnh chụp, mô tả lại bằng tiếng việt các lựa chọn (và cả các chỗ không chọn)
- Save pptx và save as sang pdf để tiện việc đọc online.
- Đăng ký tài khoản github ( sử tài khoản dụng @gmail , đừng sử dụng @tnut ), tạo repository mới, để quyền truy cập public, upload cả file pptx và pdf lên đó.
- update link của repository vào file excel online của thầy (thầy gửi link sau, vì github ko fake đc datetime) .

Chú ý:

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

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

Link bài tập:

<https://github.com/Shiwaru2004/Nguyen-Long/blob/main/C%C3%A1c%20b%C6%B0%E1%BB%9Bc%20download%20SQL%20SEVER%202022%20V%C3%80%20SQL.pdf>

## **BÀI TẬP VỀ NHÀ 02 - MÔN HỆ QUẢN TRỊ CSDL:**

1. Đã cài đặt SQL Server 2022 Dev.
2. Đã cài đặt SQL Management Studio bản mới nhất.
3. Đã kết nối từ SQL Management 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 họa 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 họa trên thành lệnh SQL tương đương. lưu tất cả các lệnh SQL trong file: Script\_DML.sql

---

## HÌNH THÚC 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).

Bài làm:

File README:

### **BAI\_TAP\_2: Nguyen\_Long**

#MSSV: K225480106043 CREATE DATABASE QLSV GO

USE QLSV

CREATE TABLE SinhVien( masv CHAR(13) NOT NULL,

hoten NVARCHAR (30) NULL,

NGAYSINH VARCHAR(10) NULL,

CONSTRAINT pk\_SinhVien PRIMARY KEY(masv), )

CREATE TABLE Lop(

maLop CHAR(6) NOT NULL,

tenlop NVARCHAR(100) NULL,

CONSTRAINT pk\_Lop PRIMARY KEY(maLop), )

CREATE TABLE GVCN(

maLop CHAR(6) NOT NULL,

magv CHAR(6) NOT NULL,

HK VARCHAR(1) NOT NULL,

CONSTRAINT pk\_GVCN PRIMARY KEY(maLop),

CONSTRAINT pk\_GVCN PRIMARY KEY(magv),

---

CONSTRAINT fk\_Lop\_GVCN FOREIGN KEY(maLop) REFERENCES  
Lop(maLop),

CONSTRAINT fk\_GiaoVien\_GVCN FOREIGN KEY (magv) REFERENCES  
GiaoVien(magv), )

CREATE TABLE LopSV(  
maLop CHAR(6) NOT NULL,  
maSV CHAR(6) NOT NULL,  
ChucVu VARCHAR(50) NULL,  
CONSTRAINT pk\_LopSV PRIMARY KEY (maLop),  
CONSTRAINT pk\_LopSV PRIMARY KEY (maSV),

CONSTRAINT fk\_Lop\_LopSV FOREIGN KEY (maLop) REFERENCES  
Lop(maLop),

CONSTRAINT fk\_SinhVien\_LopSV FOREIGN KEY (maSV) REFERENCES  
SinhVien(masv), )

CREATE TABLE GiaoVien (br/>  
magv CHAR(6) NOT NULL,  
hoten NVARCHAR(30) NOT NULL,  
NgaySinh DATE NULL,  
maBM NCHAR(6) NULL,  
CONSTRAINT pk\_GiaoVien PRIMARY KEY (magv),

CONSTRAINT fk\_BoMon\_GiaoVien FOREIGN KEY (maBM) REFERENCES  
BoMon(MaBM), );

CREATE TABLE BoMon (br/>  
MaBM NCHAR(6) NOT NULL,  
tenBM NVARCHAR(100) NULL,  
maKhoa NCHAR(7) NULL,  
CONSTRAINT pk\_BoMon PRIMARY KEY (MaBM),  
CONSTRAINT fk\_Khoa\_BoMon FOREIGN KEY (maKhoa) REFERENCES  
Khoa(maKhoa), );

---

```
CREATE TABLE Khoa (
    maKhoa NCHAR(7) NOT NULL,
    tenKhoa NVARCHAR(100)
    CONSTRAINT pk_Khoa PRIMARY KEY (maKhoa), );

CREATE TABLE MonHoc (
    mamon NCHAR(6) NOT NULL,
    Tenmon NVARCHAR(100),
    STC CHAR(3) NOT NULL, );

CREATE TABLE LopHP (
    maLopHP CHAR(6) NOT NULL,
    TenLopHP NVARCHAR(100) NULL,
    HK VARCHAR(10) NULL,
    maMon VARCHAR(6) NULL,
    maGV CHAR(6) NULL,
    CONSTRAINT pk_LopHP PRIMARY KEY (maLopHP),
    CONSTRAINT fk_MonHoc_LopHP FOREIGN KEY (maMon) REFERENCES
    MonHoc(mamon),
    CONSTRAINT fk_GiaoVien_LopHP FOREIGN KEY (maGV) REFERENCES
    GiaoVien(magv), );

CREATE TABLE DKMH (
    maLopHP CHAR(6) NOT NULL,
    maSV CHAR(13) NOT NULL,
    DiemTP INT NULL,
    DiemThi INT NULL,
    PhanTramThi NCHAR(10) NOT NULL,
    CONSTRAINT pk_DKMH PRIMARY KEY (maLopHP),
    CONSTRAINT pk_DKMH PRIMARY KEY (maSV),
```

---

---

CONSTRAINT fk\_LopHP\_DKMH FOREIGN KEY (maLopHP) REFERENCES LopHP(maLopHP),

CONSTRAINT fk\_SinhVien\_DKMH FOREIGN KEY (maSV) REFERENCES SinhVien(masv), );

File Script:

CREATE DATABASE QLSV

GO

USE QLSV

CREATE TABLE SinhVien(

masv CHAR(13) NOT NULL,

hoten NVARCHAR (30) NULL,

NGAYSINH VARCHAR(10) NULL,

CONSTRAINT pk\_SinhVien PRIMARY KEY(masv),

)

CREATE TABLE Lop(

maLop CHAR(6) NOT NULL,

tenlop NVARCHAR(100) NULL,

CONSTRAINT pk\_Lop PRIMARY KEY(maLop),

)

---

```
CREATE TABLE GVCN(
    maLop CHAR(6) NOT NULL,
    magv CHAR(6) NOT NULL,
    HK VARCHAR(1) NOT NULL,
    CONSTRAINT pk_GVCN PRIMARY KEY(maLop,magv),
    CONSTRAINT fk_Lop_GVCN FOREIGN KEY(maLop) REFERENCES
    Lop(maLop),
    CONSTRAINT fk_GiaoVien_GVCN FOREIGN KEY (magv) REFERENCES
    GiaoVien(magv),
)
```

```
CREATE TABLE LopSV(
    maLop CHAR(6) NOT NULL,
    maSV CHAR(6) NOT NULL,
    ChucVu VARCHAR(50) NULL,
    CONSTRAINT pk_LopSV PRIMARY KEY (maLop,maSV),
    CONSTRAINT fk_Lop_LopSV FOREIGN KEY (maLop) REFERENCES
    Lop(maLop),
    CONSTRAINT fk_SinhVien_LopSV FOREIGN KEY (maSV) REFERENCES
    SinhVien(masv),
)
```

```
CREATE TABLE GiaoVien (
    magv CHAR(6) NOT NULL,
```

---

```
hoten NVARCHAR(30) NOT NULL,  
NgaySinh DATE NULL,  
maBM NCHAR(6) NULL,  
CONSTRAINT pk_GiaoVien PRIMARY KEY (magv),  
CONSTRAINT fk_BoMon_GiaoVien FOREIGN KEY (maBM) REFERENCES  
BoMon(MaBM),  
);
```

```
CREATE TABLE BoMon (  
MaBM NCHAR(6) NOT NULL,  
tenBM NVARCHAR(100) NULL,  
maKhoa NCHAR(7) NULL,  
CONSTRAINT pk_BoMon PRIMARY KEY (MaBM),  
CONSTRAINT fk_Khoa_BoMon FOREIGN KEY (maKhoa) REFERENCES  
Khoa(maKhoa),  
);
```

```
CREATE TABLE Khoa (  
maKhoa NCHAR(7) NOT NULL,  
tenKhoa NVARCHAR(100)  
CONSTRAINT pk_Khoa PRIMARY KEY (maKhoa),  
);
```

---

```
CREATE TABLE MonHoc (
    mamon NCHAR(6) NOT NULL,
    Tenmon NVARCHAR(100),
    STC CHAR(3) NOT NULL,
);
```

```
CREATE TABLE LopHP (
    maLopHP CHAR(6) NOT NULL,
    TenLopHP NVARCHAR(100) NULL,
    HK VARCHAR(10) NULL,
    maMon VARCHAR(6) NULL,
    maGV CHAR(6) NULL,
    CONSTRAINT pk_LopHP PRIMARY KEY (maLopHP),
    CONSTRAINT fk_MonHoc_LopHP FOREIGN KEY (maMon) REFERENCES
        MonHoc(mamon),
    CONSTRAINT fk_GiaoVien_LopHP FOREIGN KEY (maGV) REFERENCES
        GiaoVien(magv),
);
```

```
CREATE TABLE DKMH (
    maLopHP CHAR(6) NOT NULL,
    maSV CHAR(13) NOT NULL,
```

---

---

```
DiemTP INT NULL,
DiemThi INT NULL,
PhanTramThi NCHAR(10) NOT NULL,
CONSTRAINT pk_DKMH PRIMARY KEY (maLopHP,maSV),
CONSTRAINT fk_LopHP_DKMH FOREIGN KEY (maLopHP) REFERENCES
LopHP(maLopHP),
CONSTRAINT fk_SinhVien_DKMH FOREIGN KEY (maSV) REFERENCES
SinhVien(masv),
);
```

### BÀI TẬP VỀ NHÀ 03 - MÔN HỆ QUẢN TRỊ CSDL:

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 THÚC 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 Scripts => sinh ra file: bai\_tap\_3\_schema.sql (chỉ chứa lệnh tạo cấu trúc của db)

5. dùng tính năng: Tasks -> Generate Scripts => 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

### File bai\_tap\_3\_schema.sql

USE [master]

GO

\*\*\*\*\* Object: Database [QLSV] Script Date: 3/30/2025 11:39:52 PM \*\*\*\*\*

CREATE DATABASE [QLSV]

---

---

```
CONTAINMENT = NONE
ON PRIMARY
( NAME = N'QLSV', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL16.MSSQLSERVER\MSSQL\DATA\QLSV.mdf' , SIZE =
8192KB , MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )
LOG ON
( NAME = N'QLSV_log', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL16.MSSQLSERVER\MSSQL\DATA\QLSV_log.ldf' , SIZE =
8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB )
WITH CATALOG_COLLATION = DATABASE_DEFAULT, LEDGER = OFF
GO
ALTER DATABASE [QLSV] SET COMPATIBILITY_LEVEL = 160
GO
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [QLSV].[dbo].[sp_fulltext_database] @action = 'enable'
end
GO
ALTER DATABASE [QLSV] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [QLSV] SET ANSI_NULLS OFF
GO
ALTER DATABASE [QLSV] SET ANSI_PADDING OFF
GO
ALTER DATABASE [QLSV] SET ANSI_WARNINGS OFF
GO
```

---

---

```
ALTER DATABASE [QLSV] SET ARITHABORT OFF
GO
ALTER DATABASE [QLSV] SET AUTO_CLOSE OFF
GO
ALTER DATABASE [QLSV] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [QLSV] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [QLSV] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [QLSV] SET CURSOR_DEFAULT GLOBAL
GO
ALTER DATABASE [QLSV] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [QLSV] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [QLSV] SET QUOTED_IDENTIFIER OFF
GO
ALTER DATABASE [QLSV] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [QLSV] SET ENABLE_BROKER
GO
ALTER DATABASE [QLSV] SET AUTO_UPDATE_STATISTICS_ASYNC OFF
GO
```

---

---

```
ALTER DATABASE [QLSV] SET DATE_CORRELATION_OPTIMIZATION
OFF

GO

ALTER DATABASE [QLSV] SET TRUSTWORTHY OFF

GO

ALTER DATABASE [QLSV] SET ALLOW_SNAPSHOT_ISOLATION OFF

GO

ALTER DATABASE [QLSV] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [QLSV] SET READ_COMMITTED_SNAPSHOT OFF

GO

ALTER DATABASE [QLSV] SET HONOR_BROKER_PRIORITY OFF

GO

ALTER DATABASE [QLSV] SET RECOVERY FULL

GO

ALTER DATABASE [QLSV] SET MULTI_USER

GO

ALTER DATABASE [QLSV] SET PAGE_VERIFY CHECKSUM

GO

ALTER DATABASE [QLSV] SET DB_CHAINING OFF

GO

ALTER DATABASE [QLSV] SET
FILESTREAM( NON_TRANSACTED_ACCESS = OFF )

GO

ALTER DATABASE [QLSV] SET TARGET_RECOVERY_TIME = 60
SECONDS
```

---

---

GO

ALTER DATABASE [QLSV] SET DELAYED\_DURABILITY = DISABLED

GO

ALTER DATABASE [QLSV] SET ACCELERATED\_DATABASE\_RECOVERY = OFF

GO

EXEC sys.sp\_db\_vardecimal\_storage\_format N'QLSV', N'ON'

GO

ALTER DATABASE [QLSV] SET QUERY\_STORE = ON

GO

ALTER DATABASE [QLSV] 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 [QLSV]

GO

\*\*\*\*\* Object: Table [dbo].[BoMon] Script Date: 3/30/2025 11:39:52 PM\*\*\*\*\*

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[BoMon](

    [MaBM] [nchar](10) NOT NULL,

---

```
[tenBM] [nvarchar](50) NULL,  
[maKhoa] [nchar](10) NULL,  
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: 3/30/2025 11:39:52 PM  
*****/  
SET ANSI_NULLS ON  
GO  
SET QUOTED_IDENTIFIER ON  
GO  
CREATE TABLE [dbo].[Diem](  
    [id] [nchar](10) NOT NULL,  
    [id_dk] [nchar](10) NULL,  
    [Diem] [int] 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: 3/30/2025 11:39:52 PM  
\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[DKMH](

    [maLopHP] [nchar](10) NOT NULL,

    [maSV] [nchar](13) NOT NULL,

    [DiemThi] [int] NULL,

    [PhamTramThi] [nchar](10) NOT 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: 3/30/2025 11:39:52 PM  
\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

---

---

```
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[GiaoVien](
    [magv] [nchar](10) NOT NULL,
    [hoten] [nvarchar](50) NOT NULL,
    [NgaySinh] [date] NULL,
    [maBM] [nchar](10) NULL,
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: 3/30/2025 11:39:52 PM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[GVCN](
    [maLop] [nchar](10) NOT NULL,
    [magv] [nchar](10) NOT NULL,
    [HK] [nchar](10) NOT NULL,
```

---

PRIMARY KEY CLUSTERED

(

    [maLop] ASC,

    [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].[Khoa] Script Date: 3/30/2025 11:39:52 PM  
\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Khoa](

    [maKhoa] [nchar](10) NOT NULL,

    [tenKhoa] [nvarchar](50) NULL,

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].[lop]  Script Date: 3/30/2025 11:39:52 PM *****
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE TABLE [dbo].[lop](
```

```
    [maLop] [nchar](10) NOT NULL,
```

```
    [tenLop] [nchar](20) NULL,
```

```
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: 3/30/2025 11:39:52 PM  
*****
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE TABLE [dbo].[LopHP](
```

```
    [maLopHP] [nchar](10) NOT NULL,
```

```
    [TenLopHP] [nvarchar](50) NULL,
```

---

```
[HK] [nchar](10) NULL,
[maMon] [nchar](10) NULL,
[maGV] [nchar](10) NULL,
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: 3/30/2025 11:39:52 PM *****/
SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

CREATE TABLE [dbo].[LopSV](
[maLop] [nchar](10) NOT NULL,
[maSV] [nchar](13) NOT NULL,
[ChucVu] [nvarchar](30) NULL,
PRIMARY KEY CLUSTERED
(
[maLop] ASC,
[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: 3/30/2025 11:39:52 PM
*****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE TABLE [dbo].[MonHoc](
```

```
    [mamon] [nchar](10) NOT NULL,
```

```
    [Tenmon] [nvarchar](50) NULL,
```

```
    [STC] [int] NOT NULL,
```

```
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: 3/30/2025 11:39:52 PM
*****/
```

```
SET ANSI_NULLS ON
```

---

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[SinhVien](

    [masv] [nchar](13) NOT NULL,

    [hoten] [nvarchar](50) NULL,

    [Ngaysinh] [date] NULL,

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 FOREIGN KEY([maKhoa])  
REFERENCES [dbo].[Khoa] ([maKhoa])

GO

ALTER TABLE [dbo].[Diem] WITH CHECK ADD CONSTRAINT  
[FK\_Diem\_Diem1] FOREIGN KEY([id])  
REFERENCES [dbo].[Diem] ([id])

GO

ALTER TABLE [dbo].[Diem] CHECK CONSTRAINT [FK\_Diem\_Diem1]

GO

ALTER TABLE [dbo].[DKMH] WITH CHECK ADD CONSTRAINT  
[FK\_DKMH\_maLopHP\_5070F446] FOREIGN KEY([maLopHP])

---

---

REFERENCES [dbo].[LopHP] ([maLopHP])

GO

ALTER TABLE [dbo].[DKMH] CHECK CONSTRAINT  
[FK\_DKMH\_maLopHP\_5070F446]

GO

ALTER TABLE [dbo].[DKMH] WITH CHECK ADD CONSTRAINT  
[FK\_DKMH\_maSV\_5165187F] FOREIGN KEY([maSV])

REFERENCES [dbo].[SinhVien] ([masv])

GO

ALTER TABLE [dbo].[DKMH] CHECK CONSTRAINT  
[FK\_DKMH\_maSV\_5165187F]

GO

ALTER TABLE [dbo].[GiaoVien] WITH CHECK ADD FOREIGN KEY([maBM])

REFERENCES [dbo].[BoMon] ([MaBM])

GO

ALTER TABLE [dbo].[GVCN] WITH CHECK ADD FOREIGN KEY([magv])

REFERENCES [dbo].[GiaoVien] ([magv])

GO

ALTER TABLE [dbo].[GVCN] WITH CHECK ADD FOREIGN KEY([maLop])

REFERENCES [dbo].[lop] ([maLop])

GO

ALTER TABLE [dbo].[LopHP] WITH CHECK ADD FOREIGN KEY([maGV])

REFERENCES [dbo].[GiaoVien] ([magv])

GO

ALTER TABLE [dbo].[LopHP] WITH CHECK ADD FOREIGN KEY([maMon])

REFERENCES [dbo].[MonHoc] ([mamon])

---

---

GO

ALTER TABLE [dbo].[LopSV] WITH CHECK ADD FOREIGN KEY([maLop])  
REFERENCES [dbo].[lop] ([maLop])

GO

ALTER TABLE [dbo].[LopSV] WITH CHECK ADD FOREIGN KEY([maSV])  
REFERENCES [dbo].[SinhVien] ([masv])

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].[GVCN] WITH CHECK ADD CONSTRAINT  
[CK\_GVCN\_HK] CHECK (([HK]>(0) AND [HK]<=(4)))

GO

ALTER TABLE [dbo].[GVCN] CHECK CONSTRAINT [CK\_GVCN\_HK]

GO

ALTER TABLE [dbo].[LopHP] WITH CHECK ADD CONSTRAINT  
[CK\_LopHP\_HK] CHECK (([HK]>(0) AND [HK]<=(4)))

GO

ALTER TABLE [dbo].[LopHP] CHECK CONSTRAINT [CK\_LopHP\_HK]

GO

ALTER TABLE [dbo].[MonHoc] WITH CHECK ADD CONSTRAINT  
[CK\_MonHoc\_STC] CHECK (([STC]>(0) AND [STC]<=(4)))

GO

ALTER TABLE [dbo].[MonHoc] CHECK CONSTRAINT [CK\_MonHoc\_STC]

---

---

GO

ALTER TABLE [dbo].[SinhVien] WITH CHECK ADD CONSTRAINT  
[CK\_SinhVien\_NgaySinh] CHECK (([NgaySinh]<='2007-01-01'))

GO

ALTER TABLE [dbo].[SinhVien] CHECK CONSTRAINT  
[CK\_SinhVien\_NgaySinh]

GO

USE [master]

GO

ALTER DATABASE [QLSV] SET READ\_WRITE

GO

File bai\_tap\_3\_data.sql:

USE [QLSV]

GO

INSERT [dbo].[SinhVien] ([masv], [hoten], [Ngaysinh]) VALUES  
(N'k225480106034', N'ven long', CAST(N'2002-02-02' AS Date))

INSERT [dbo].[SinhVien] ([masv], [hoten], [Ngaysinh]) VALUES  
(N'k225480106043', N'nguyen long', CAST(N'2004-01-12' AS Date))

INSERT [dbo].[SinhVien] ([masv], [hoten], [Ngaysinh]) VALUES  
(N'k225480106054', N'long nguyen', CAST(N'2001-01-01' AS Date))

GO

INSERT [dbo].[Khoa] ([maKhoa], [tenKhoa]) VALUES (N'khoa01', N'dien tu')

GO

INSERT [dbo].[BoMon] ([MaBM], [tenBM], [maKhoa]) VALUES (N'bm01',  
N'a', N'khoa01')

GO

INSERT [dbo].[GiaoVien] ([magv], [hoten], [NgaySinh], [maBM]) VALUES  
(N'gv01', N'gv a', CAST(N'1999-01-01' AS Date), N'bm01')

---

---

GO

INSERT [dbo].[MonHoc] ([mamon], [Tenmon], [STC]) VALUES (N'mh01 ', N'toan', 3)

GO

INSERT [dbo].[LopHP] ([maLopHP], [TenLopHP], [HK], [maMon], [maGV]) VALUES (N'l01 ', N'toan 2 ', N'3 ', N'mh01 ', N'gv01 ')

GO

INSERT [dbo].[DKMH] ([maLopHP], [maSV], [DiemThi], [PhamTramThi], [id\_dk]) VALUES (N'l01 ', N'k225480106043', 10, N'0.7 ', N'1 ')

INSERT [dbo].[DKMH] ([maLopHP], [maSV], [DiemThi], [PhamTramThi], [id\_dk]) VALUES (N'l01 ', N'k225480106054', 9, N'0.7 ', N'2 ')

INSERT [dbo].[DKMH] ([maLopHP], [maSV], [DiemThi], [PhamTramThi], [id\_dk]) VALUES (N'l01 ', N'k225480106034', 8, N'0.7 ', N'3 ')

GO

INSERT [dbo].[Diem] ([id], [id\_dk], [Diem]) VALUES (N'1 ', N'1 ', 4)

INSERT [dbo].[Diem] ([id], [id\_dk], [Diem]) VALUES (N'2 ', N'1 ', 5)

INSERT [dbo].[Diem] ([id], [id\_dk], [Diem]) VALUES (N'3 ', N'3 ', 3)

INSERT [dbo].[Diem] ([id], [id\_dk], [Diem]) VALUES (N'4 ', N'5 ', 6)

INSERT [dbo].[Diem] ([id], [id\_dk], [Diem]) VALUES (N'5 ', N'6 ', 7)

GO

Bài tập 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 tùy 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:

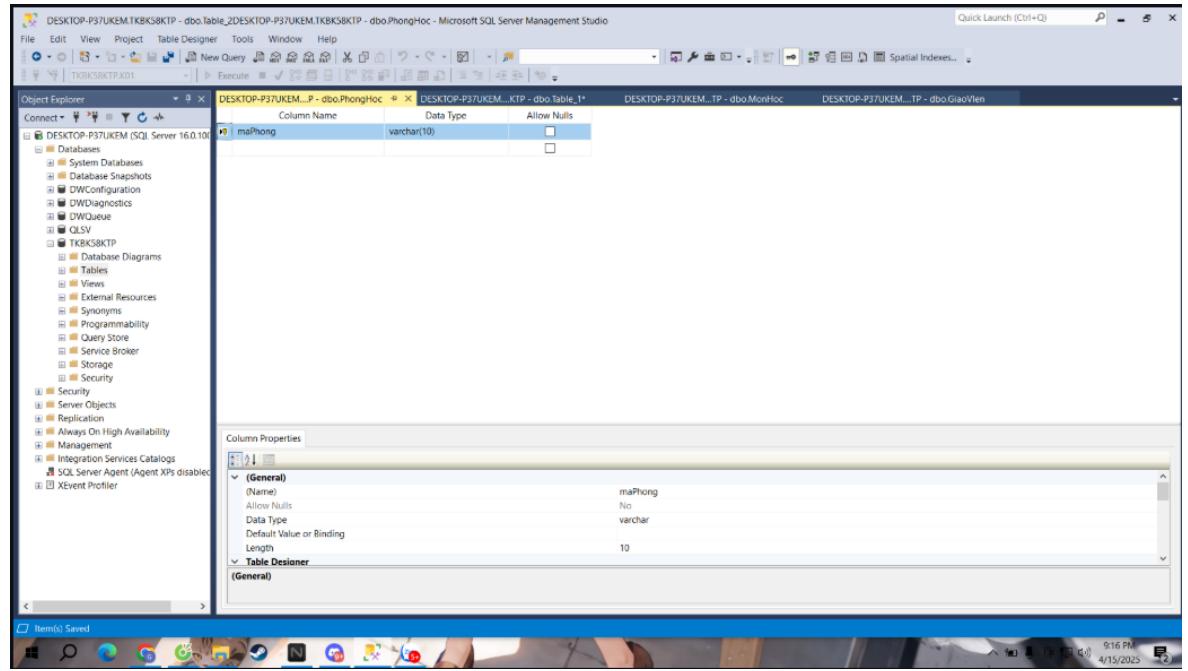
1. Tạo github repo mới: đặt tên tùy ý (có liên quan đến bài tập này)
2. tạo file readme.md, edit online nó:

paste những ảnh chụp màn hình

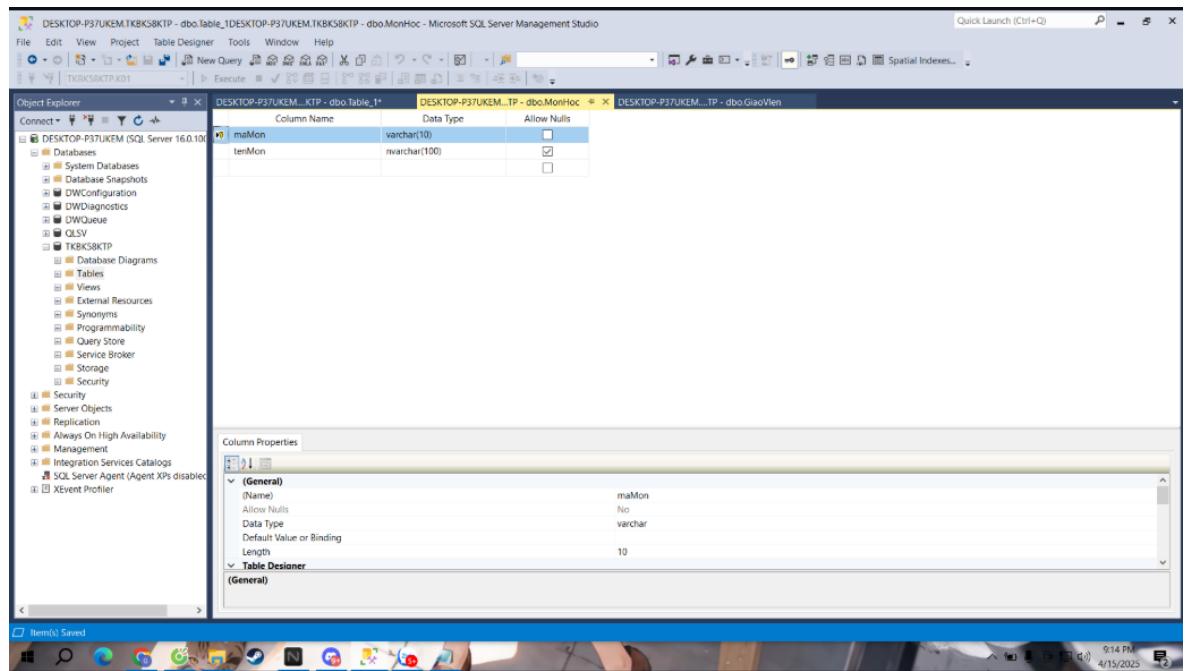
gõ text mô tả cho ảnh đó

## **bài tap 4: (sql server)**

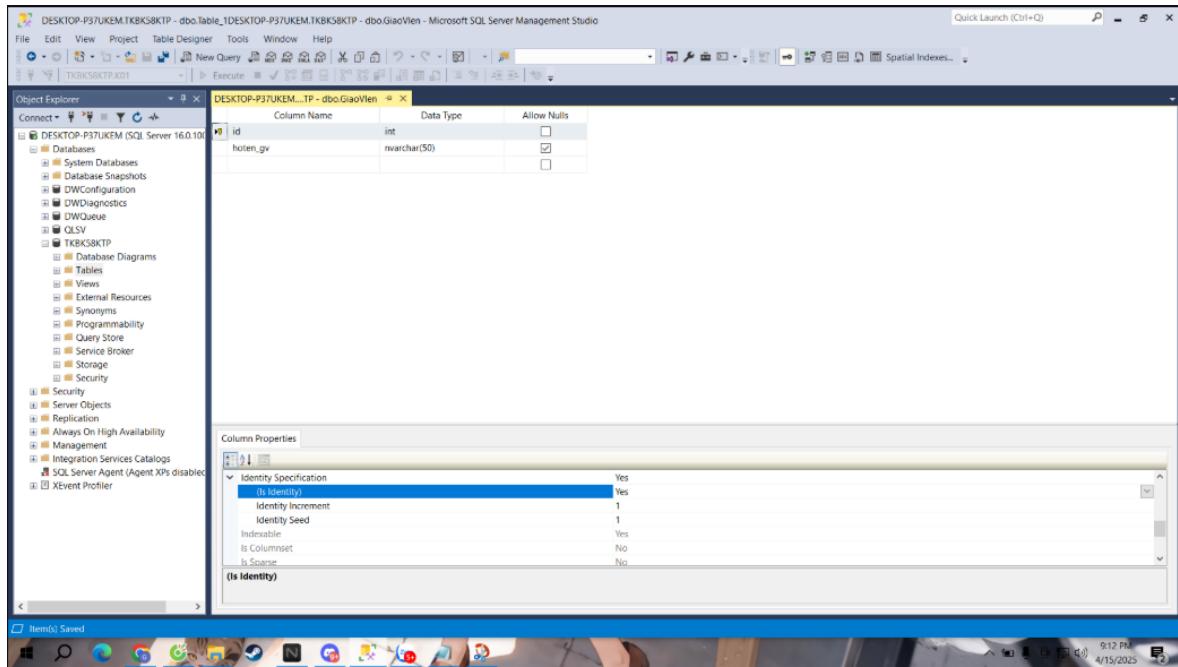
- Yêu cầu bài toán:
  - Tạo csdl cho hệ thống TKB (đã nghe giảng, đã xem cách làm)
  - Nguồn dữ liệu: TMS.tnut.edu.vn
  - Tạo các bảng tùy ý (3nf)
  - Tạo được query truy vấn ra thông tin gồm 4 cột: họ tên gv, môn dạy, giờ vào lớp, giờ ra. trả lời câu hỏi: trong khoảng thời gian từ datetime1 tới datetime2 thì có những gv nào đang bận giảng dạy.
  - Các bước thực hiện:
    1. Tạo github repo mới: đặt tên tùy 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 dụng 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 dc)
- #bài làm: Tạo các bảng cần thiết theo dữ liệu của TMS: -Bảng phòng học:



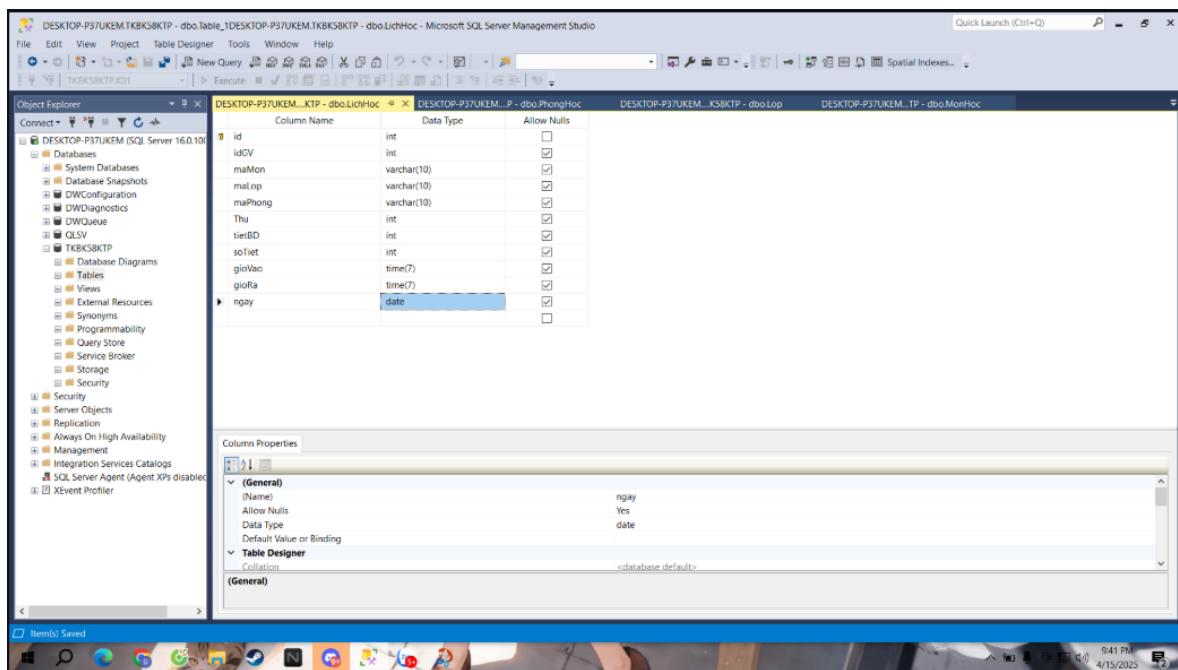
-Bảng Môn Học:



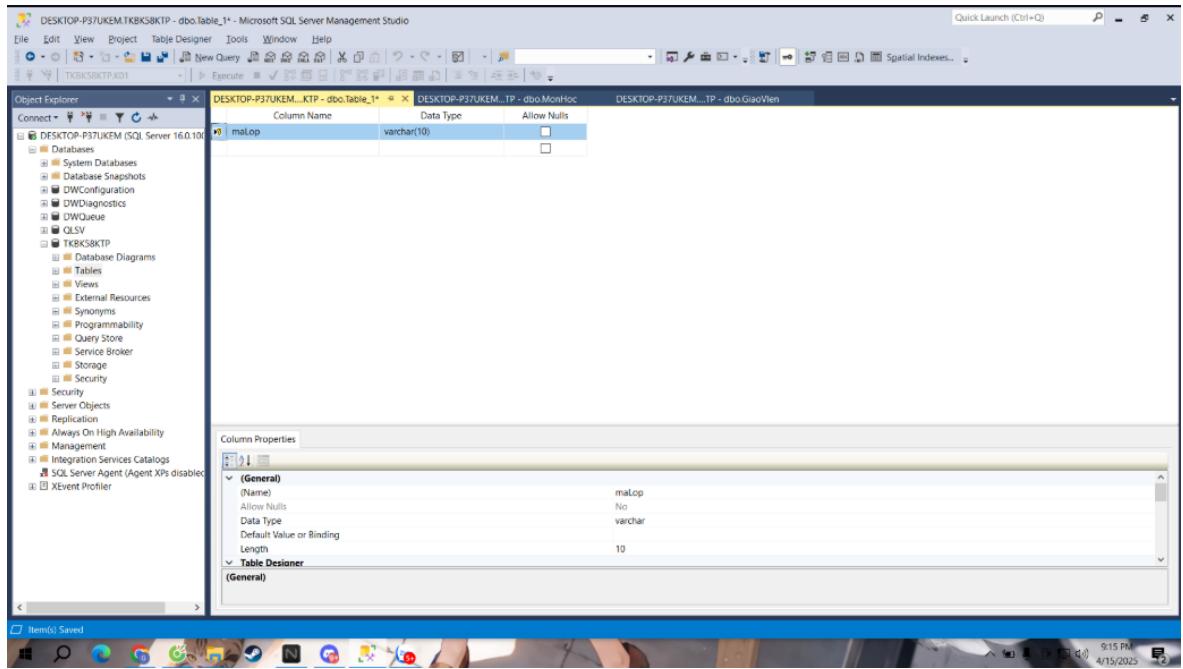
-Bảng Giáo viên



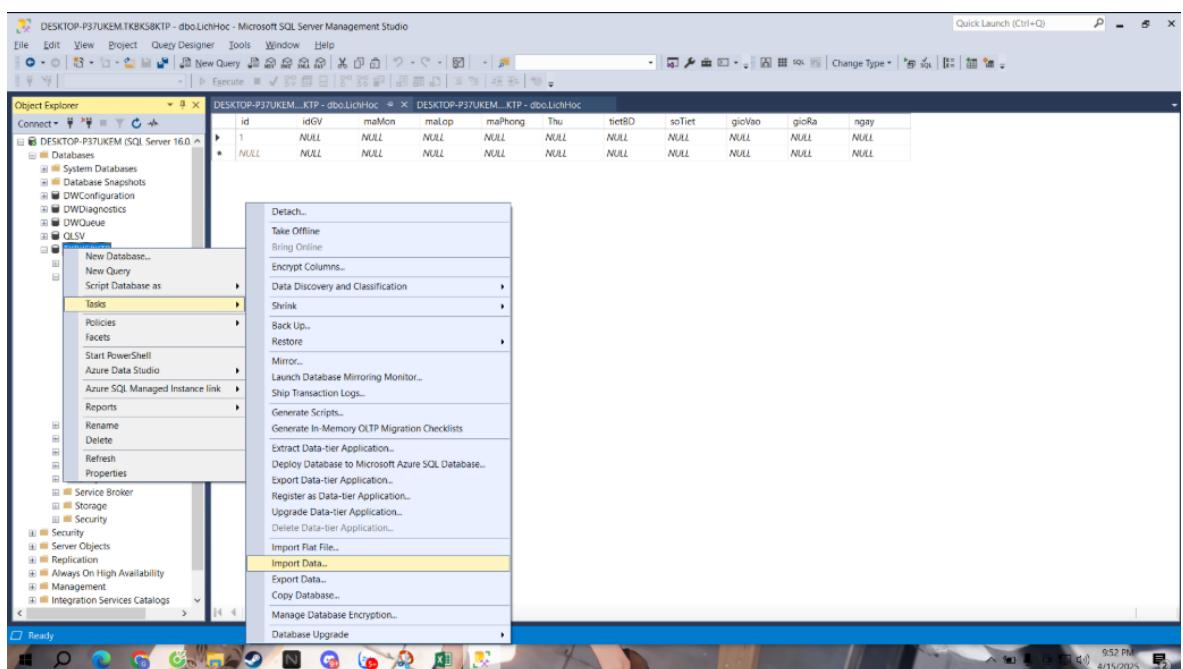
-Bảng Lịch học ( đây là bảng tổng hợp )



-Bảng Lớp



Thêm dữ liệu vào các bảng, có thể thêm bằng cách import task hoặc edit: Import file Excel:



-EDIT trực tiếp: +Bảng GV

DESKTOP-P37UKEM.TKBKS8KTP - dbo.GiaoVien - Microsoft SQL Server Management Studio

Object Explorer

Connect ▾

DESKTOP-P37UKEM (SQL Server 16.0.10) ▾

- Databases
  - System Databases
  - Database Snapshots
  - DWConfiguration
  - DWDiagnostics
  - DWQueue
  - QLSV
  - TKBK8KTP
    - Database Diagrams
    - Tables
      - System Tables
      - FileTables
      - External Tables
      - Graph Tables
      - dbo.GiaoVien
      - dbo.LichHoc
      - dbo.Lop
      - dbo.MonHoc
      - dbo.PhongHoc
      - Dropped Ledger Tables
    - Views
    - External Resources
    - Synonyms
    - Programmability
    - Query Store
    - Service Broker
    - Storage
    - Security
  - Security
  - Server Objects
  - Replication
  - Always On High Availability
  - Management
  - Integration Services Catalogs

MaGV hoten

1	NVTINH
2	TTTHANH
3	DDCOP
**	NULL

Ready

10:38 PM 4/15/2025

### +Bảng Lớp

DESKTOP-P37UKEM.TKBKS8KTP - dbo.Lop - Microsoft SQL Server Management Studio

Object Explorer

Connect ▾

DESKTOP-P37UKEM (SQL Server 16.0.10) ▾

- Databases
  - System Databases
  - Database Snapshots
  - DWConfiguration
  - DWDiagnostics
  - DWQueue
  - QLSV
  - TKBK8KTP
    - Database Diagrams
    - Tables
      - System Tables
      - FileTables
      - External Tables
      - Graph Tables
      - dbo.GiaoVien
      - dbo.LichHoc
      - dbo.Lop
      - dbo.MonHoc
      - dbo.PhongHoc
      - Dropped Ledger Tables
    - Views
    - External Resources
    - Synonyms
    - Programmability
    - Query Store
    - Service Broker
    - Storage
    - Security
  - Security
  - Server Objects
  - Replication
  - Always On High Availability
  - Management
  - Integration Services Catalogs

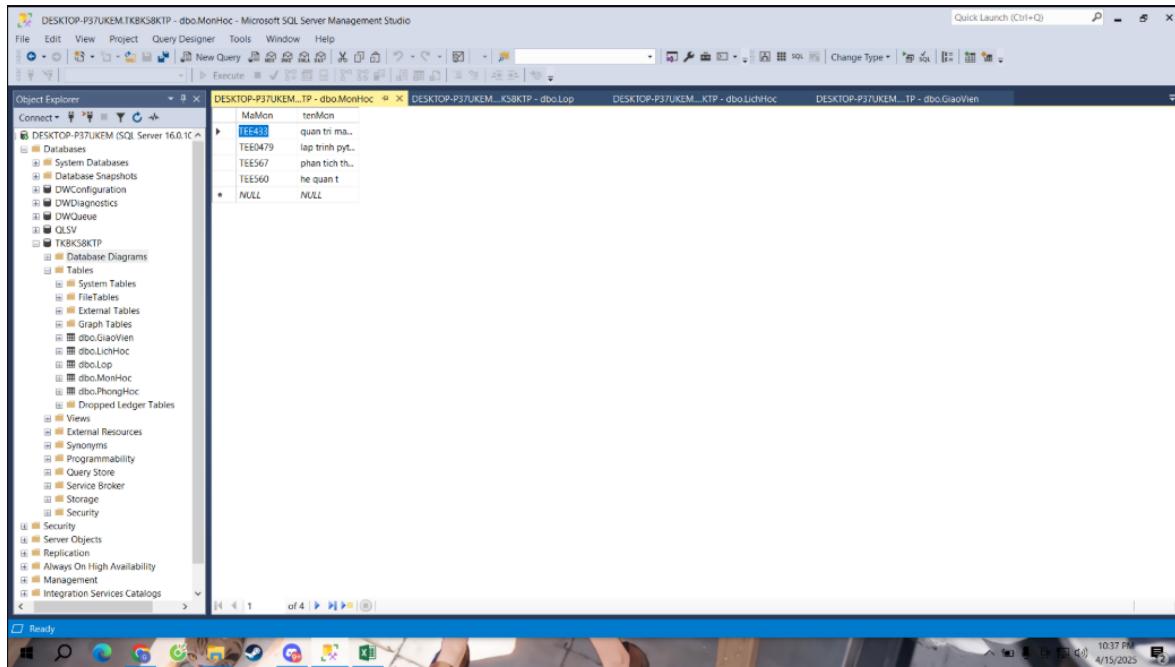
MaLop TenLop

1	K58KTP
**	NULL

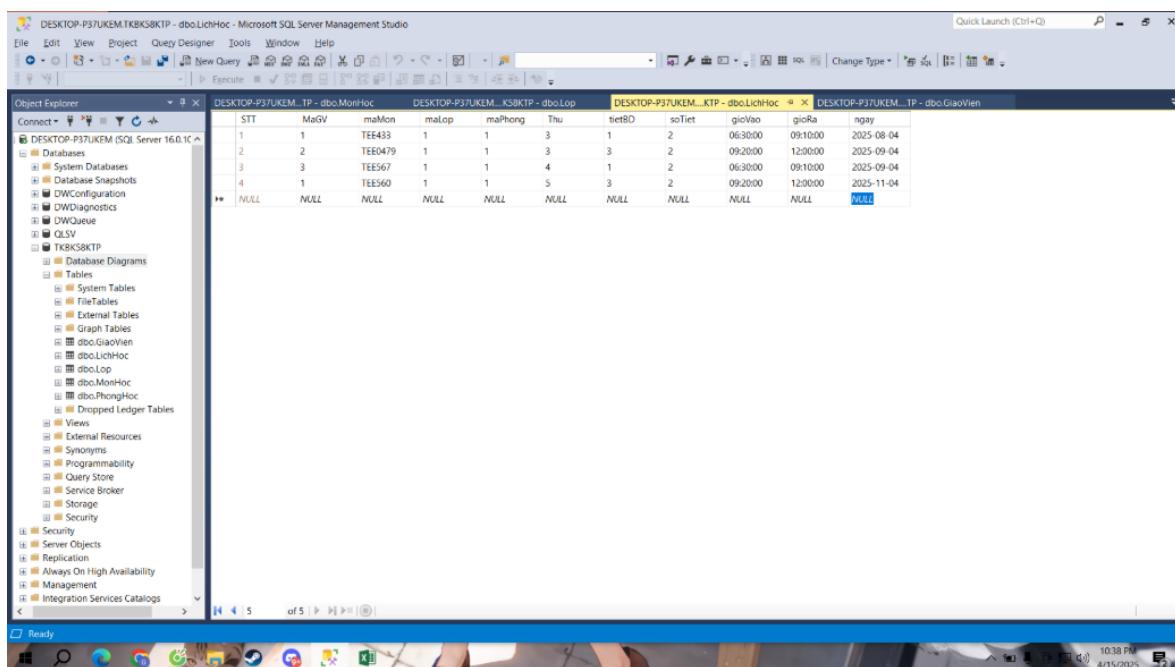
Ready

10:38 PM 4/15/2025

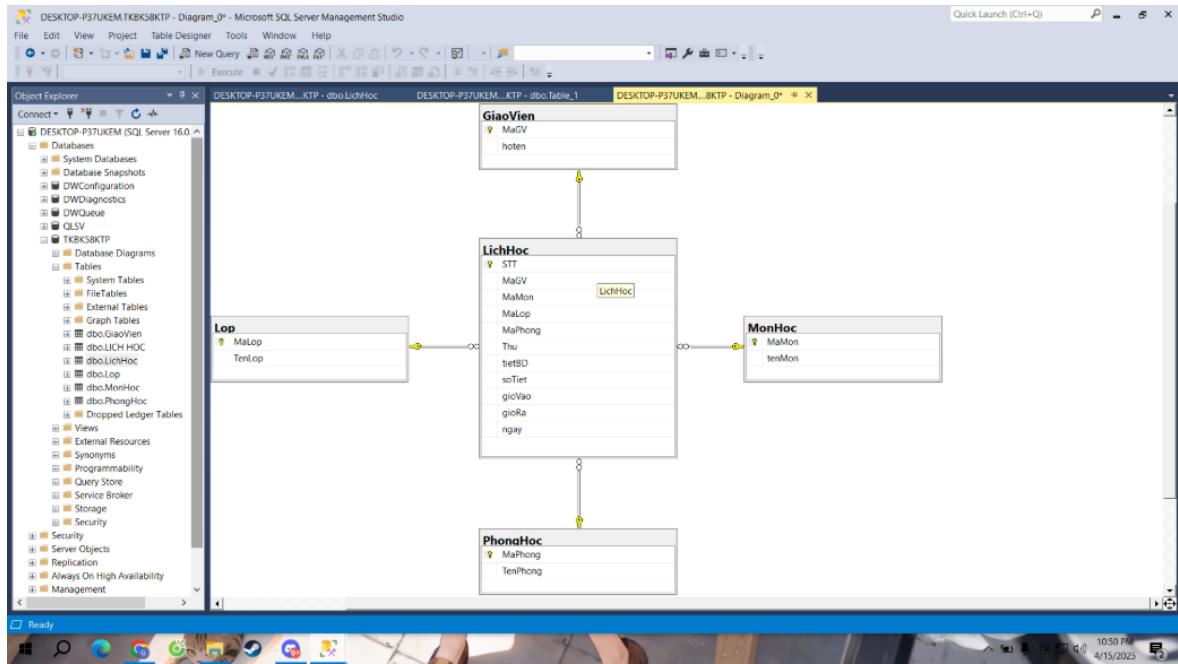
### +Bảng Môn Học



+Bảng Lịch Học:



Sau khi hoàn thành ta có sơ đồ như sau:



Truy vấn thông tin các cột và datetime: SELECT

GV.hoten AS [Họ tên GV],

MH.tenMon [Môn dạy],

CASE

WHEN Id.thu = 2 THEN 'T2'

WHEN Id.thu = 3 THEN 'T3'

WHEN Id.thu = 4 THEN 'T4'

WHEN Id.thu = 5 THEN 'T5'

WHEN Id.thu = 6 THEN 'T6'

WHEN Id.thu = 7 THEN 'T7'

WHEN Id.thu = 8 THEN 'CN'

ELSE 'N/A'

END AS [Thu],

FORMAT(Id.ngay, 'dd/MM/yyyy') AS [Ngay],

Id.so\_tiet AS [So tiet],

MIN(t.gio\_vao) AS [Gio vao lop],

---

```
MAX(t.gio_ra) AS [Gio ra lop]
FROM
LichHoc ld
INNER JOIN GiaoVien gv ON LichHoc.MaGV = gv.id
INNER JOIN MonHoc mh ON LichHoc.monhoc_id = mh.id
INNER JOIN TietHoc t ON t.tiet BETWEEN ld.tiet_bat_dau AND (ld.tiet_bat_dau +
ld.so_tiet - 1)
WHERE
ld.giang_vien_id = 4
GROUP BY
gv.ten_gv,
mh.ten_mon,
ld.thu,
ld.ngay,
ld.so_tiet,
ld.tiet_bat_dau
ORDER BY
ld.ngay, MIN(t.gio_vao);
```

## **BÀI TẬP VỀ NHÀ 05, Môn Hệ quản trị csdl.**

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 dc, 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.

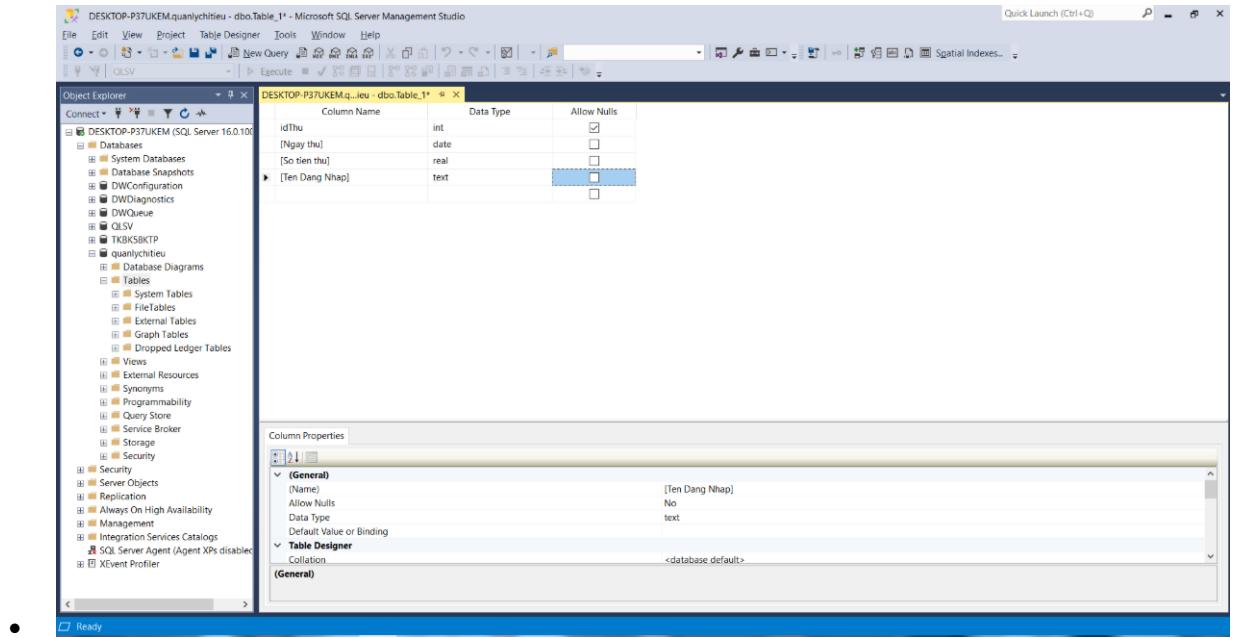
Hướng dẫn làm phần B:

- 
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 .
  3. Các thao tác làm trên csdl bằng phần mềm ssms.
  4. Chụp ảnh màn hình quá trình làm.
  5. Paste ngay vào Readme.md,  
rồi gõ mô tả ảnh này làm gì, nhập gì, hay đạt được điều gì...
  6. Có thể thêm những nhận xét hoặc kết luận  
cho việc bản thân đã hiểu rõ thêm về 1 vấn đề gì đó.
  7. Lặp lại các step 4 5 6 cho đến khi hoàn thành yêu cầu của phần B.
  8. Xuất các file sql chứa cấu trúc và data, up lên cùng repo.
  9. Link đến repo cần mở được trực tiếp nội dung,  
Paste link này vào file excel online ghim trên nhóm.

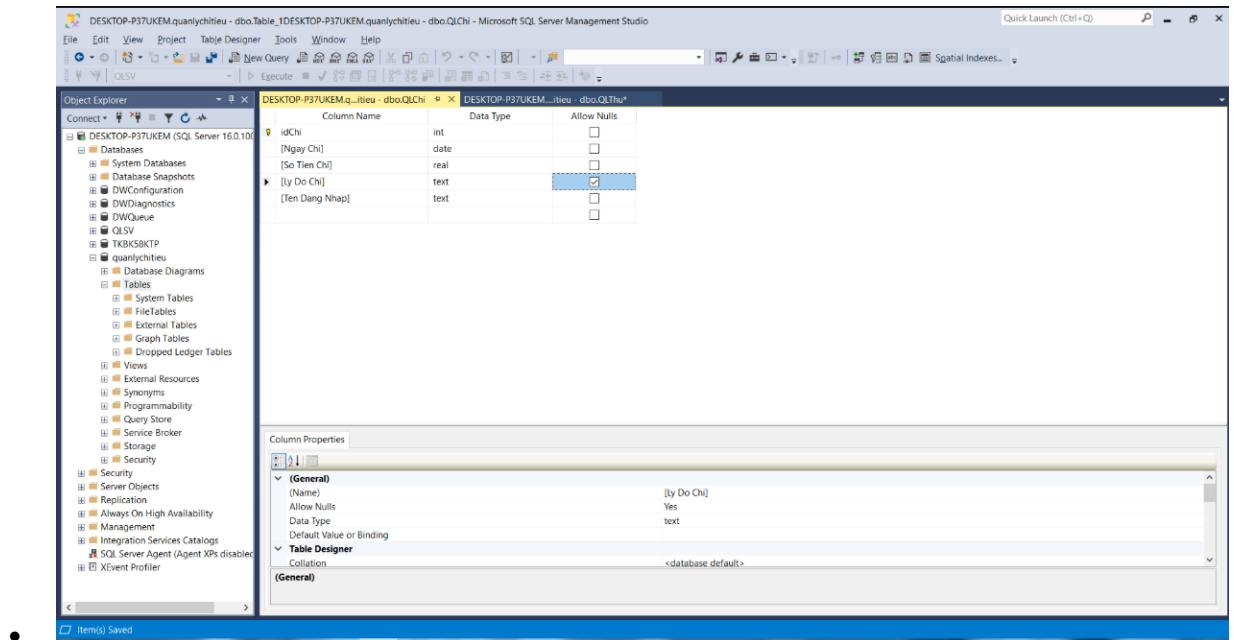
### **Bài làm:**

A. Trình bày đầu bài của đồ án PT&TKHT:

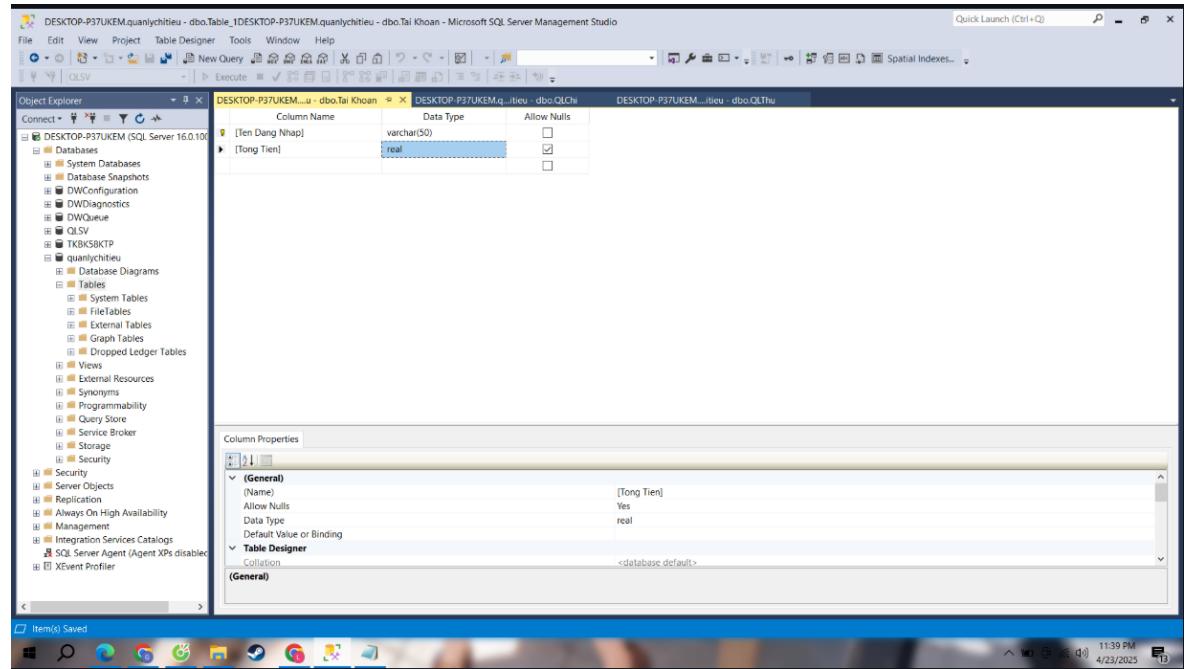
- 1 . Yêu cầu đồ án: Phân tích thiết kế phần mềm quản lý chi tiêu cá nhân
- 2 . Với những yêu cầu cần thiết về bài tập, em tạo các bảng như sau
  - Bảng Quản lý thu:



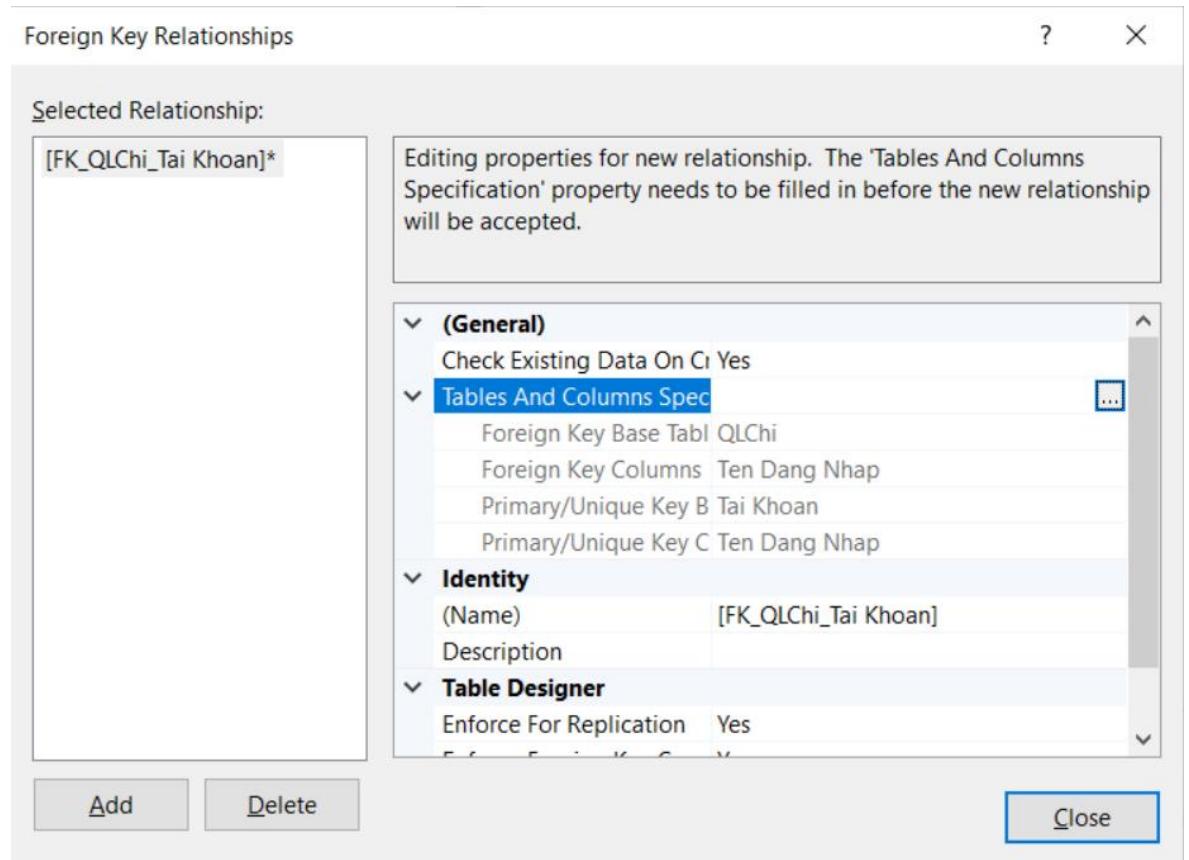
- **Bảng Quản Lý chi:**



- **Bảng Tài Khoản:**

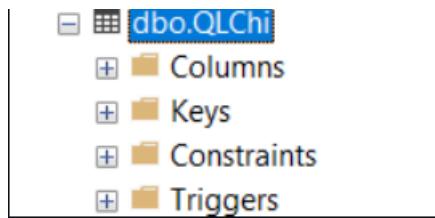


Tạo khóa ngoại liên kết cho bảng:



B.Nội dung Bài tập 05: Dựa trên CSDL của đồ án, bây giờ em cần phải set trigger để tự động cập nhật số dư (Tong Tien) Viết trigger cho các bảng để đạt được mục tiêu:

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



Trigger thêm chỉ:

```

-- Use the Specify Values for Template Parameters
-- command (Ctrl+Shift-M) to fill in the parameter
-- values below.
--
-- See additional Create Trigger templates for more
-- examples of different Trigger statements.
--
-- This block of comments will not be included in
-- the definition of the function.
=====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
=====
-- Author: Nguyen Giang Long
-- Create date: 24/4/2025
-- Description: them chi
=====
CREATE TRIGGER dbo.trg_CapNhatTongTienSauChi
ON dbo.QLChi
AFTER INSERT
AS
BEGIN
    SET NOCOUNT ON;
    UPDATE TK
    SET TK.TongTien = TK.TongTien + I.SoTienChi
    FROM dbo.TaiKhoan TK
    INNER JOIN inserted I ON TK.TenDangNhap = I.TenDangNhap;
END
GO

```

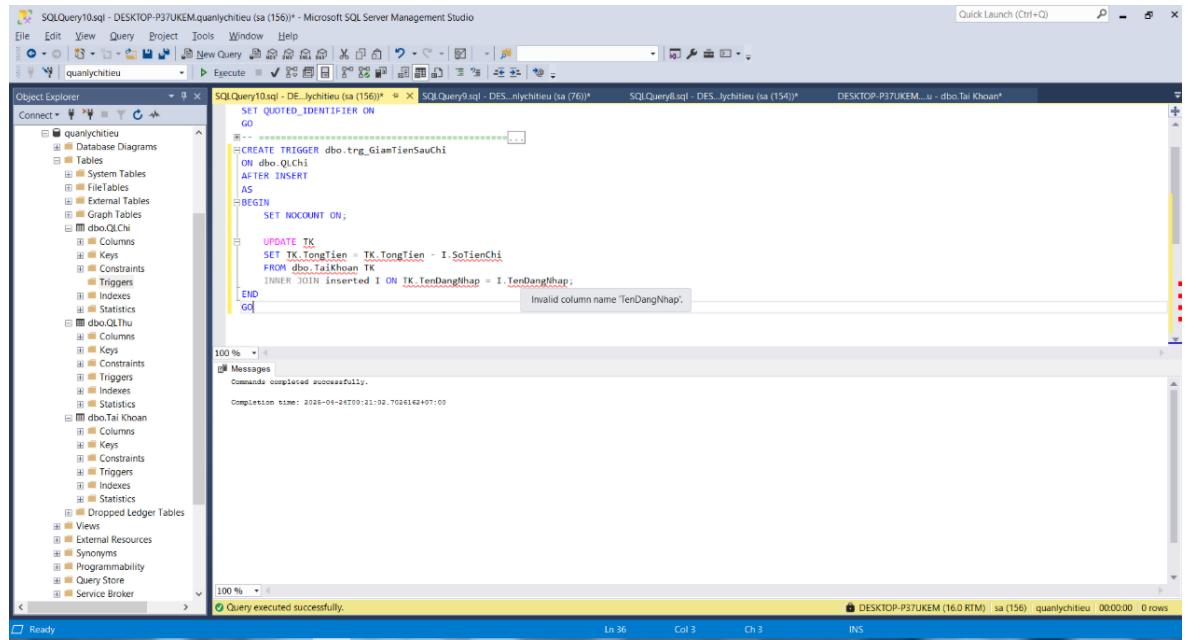
Trigger thêm thu:

```

-- Author: Nguyen Giang Long
-- Create date: 24/4/2025
-- Description: q1 thu
=====
CREATE TRIGGER dbo.trg_CapNhatTongTienSauThu
ON dbo.QLThu
AFTER INSERT
AS
BEGIN
    SET NOCOUNT ON;
    UPDATE TK
    SET TK.TongTien = TK.TongTien + I.SoTienThu
    FROM dbo.TaiKhoan TK
    INNER JOIN inserted I ON TK.TenDangNhap = I.TenDangNhap;
END
GO

```

## Trigger cho QLChi + Thêm chi(Giảm Tiền):



```
SET QUOTED_IDENTIFIER ON
GO
CREATE TRIGGER dbo.trg_GiamTienSauChi
ON dbo.QLChi
AFTER INSERT
AS
BEGIN
    SET NOCOUNT ON;

    UPDATE TK
    SET TK.TongTien = TK.TongTien - I.SoTienChi
    FROM dbo.TaiKhoan TK
    INNER JOIN inserted I ON TK.TenDangNhap = I.TenDangNhap;
END
GO
```

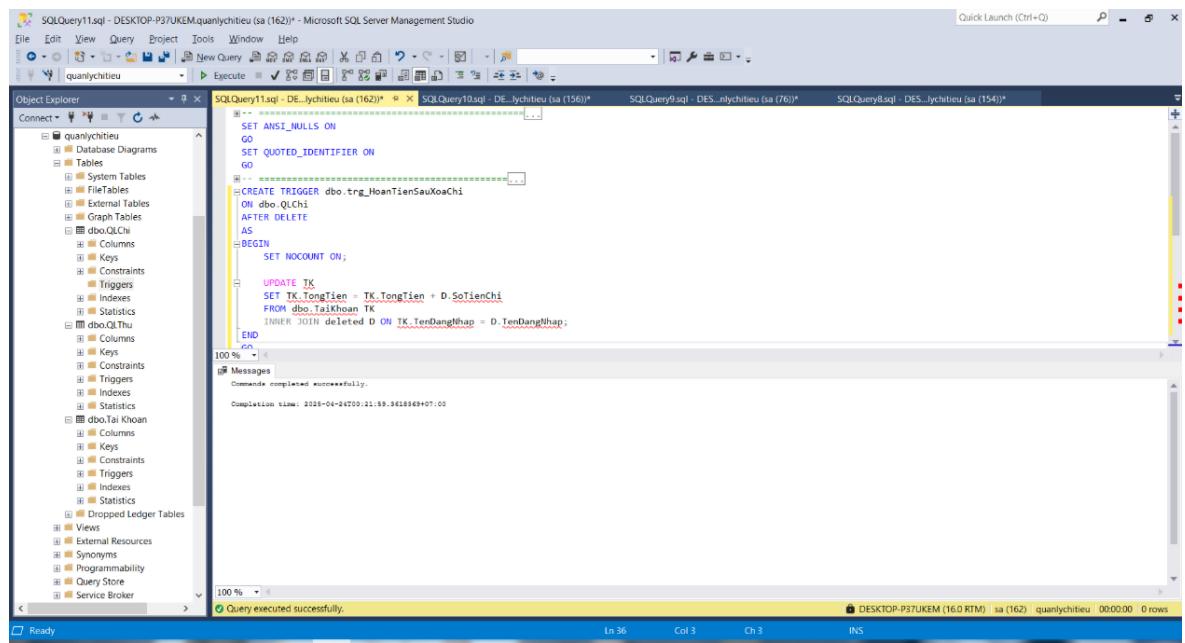
Messages

Commands completed successfully.

Completion time: 2026-04-24T09:21:02.708162+07:00

Query executed successfully.

## +Xóa chi(Hoàn tiền):



```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TRIGGER dbo.trg_HoanTienSauXoaChi
ON dbo.QLChi
AFTER DELETE
AS
BEGIN
    SET NOCOUNT ON;

    UPDATE TK
    SET TK.TongTien = TK.TongTien + D.SoTienChi
    FROM dbo.TaiKhoan TK
    INNER JOIN deleted D ON TK.TenDangNhap = D.TenDangNhap;
END
GO
```

Messages

Commands completed successfully.

Completion time: 2026-04-24T09:21:59.3618869+07:00

Query executed successfully.

## +Sửa chi(Điều chỉnh):

SQLQuery12.sql - DESKTOP-P37UKEM.qlchitieu (sa (165)) - Microsoft SQL Server Management Studio

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TRIGGER dbo.trg_DieuChinhSauCapNhatChi
ON dbo.QLChi
AFTER UPDATE
AS
BEGIN
    SET NOCOUNT ON;

    UPDATE TK
    SET TK.TongTien = TK.TongTien + D.SoTienChi - I.SoTienChi
    FROM dbo.TaiKhoan TK
    INNER JOIN inserted I ON TK.TenDangNhap = I.TenDangNhap
    INNER JOIN deleted D ON TK.TenDangNhap = D.TenDangNhap;
END

```

100 %

Messages

Commands completed successfully.

Completion time: 2026-04-24T00:13:06.4090373+07:00

Query executed successfully.

LN 37 Col 3 Ch 3 INS

DESKTOP-P37UKEM (16.0 RTM) sa (165) qlchitieu 00:00:00 0 rows

Trigger cho QLThu: +Thêm Thu(tăng tiền):

SQLQuery13.sql - DESKTOP-P37UKEM.qlchitieu (sa (76)) - Microsoft SQL Server Management Studio

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TRIGGER dbo.trg_TangTienSauThu
ON dbo.QLThu
AFTER INSERT
AS
BEGIN
    SET NOCOUNT ON;

    UPDATE TK
    SET TK.TongTien = TK.TongTien + I.SoTienThu
    FROM dbo.TaiKhoan TK
    INNER JOIN inserted I ON TK.TenDangNhap = I.TenDangNhap;
END

```

100 %

Messages

Commands completed successfully.

Completion time: 2026-04-24T00:18:04.0523284+07:00

Query executed successfully.

LN 36 Col 3 Ch 3 INS

DESKTOP-P37UKEM (16.0 RTM) sa (76) qlchitieu 00:00:00 0 rows

+Xóa thu(Giảm tiền):

SQLQuery14.sql - DESKTOP-P37UKEM.qlchieu (sa (156)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query Save All Open All Recent

Object Explorer

Connect

SQLQuery14.sql - DE...qlchieu (sa (156)\*

SQLQuery13.sql - DE...qlchieu (sa (76)\*

DESKTOP-P37UKEM...u - dbo.TaiKhoan\*

DESKTOP-P37UKEM.q...ieu - dbo.QLCh\*

quantychieu

quantychieu

Database Diagrams

Tables

System Tables

FileTables

External Tables

Graph Tables

dbo.QLChi

Columns

Keys

Constraints

Triggers

Indexes

Statistics

dbo.QLThu

Columns

Keys

Constraints

Triggers

Indexes

Statistics

dbo.TaiKhoan

Columns

Keys

Constraints

Triggers

Indexes

Statistics

Dropped Ledger Tables

Views

External Resources

Synonyms

Programmability

Query Store

Service Broker

SET ANSI\_NULLS ON  
GO  
SET QUOTED\_IDENTIFIER ON  
GO  
=====  
CREATE TRIGGER dbo.trg\_TruTienSauXoaThu  
ON dbo.QLThu  
AFTER DELETE  
AS  
BEGIN  
 SET NOCOUNT ON;  
  
 UPDATE TK  
 SET TK.TongTien = TK.TongTien - D.SoTienThu  
 FROM dbo.TaiKhoan TK  
 INNER JOIN deleted D ON TK.TenDangNhap = D.TenDangNhap;  
END  
GO

100 %

Messages

Commands completed successfully.

Completion time: 2024-04-24T20:26:53.6620000+07:00

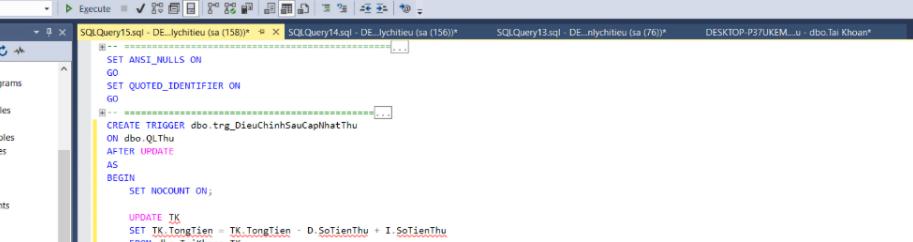
100 %

Query executed successfully.

1 in 36 Col 3 Ch 3 INS

DESKTOP-P37UKEM (16.0 RTM) | sa (156) | quantychieu | 00:00:00 | 0 rows

+ Sửa thu (Điều chỉnh):



```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET NOCOUNT ON;

CREATE TRIGGER dbo.trg_DieuChinhSoTienThu
ON dbo.QLThu
AFTER UPDATE
AS
BEGIN
    UPDATE TK
    SET TK.TongTien = TK.TongTien - D.SoTienThu + I.SoTienThu
    FROM dbo.TaiKhoan TK
    INNER JOIN inserted I ON TK.TenDangNhap = I.TenDangNhap
    INNER JOIN deleted D ON TK.TenDangNhap = D.TenDangNhap;
END
```

100 %

Messages

Commands completed successfully.

Completion time: 2018-04-24T00:26:37.8118978+07:00

Test trigger: Ban đầu:

SQLQuery15.sql - DESKTOP-P37UKEM.qlchitieu (sa (158)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Object Explorer

SQLQuery15.sql - DE..lychitieu (sa (158)) \* SQLQuery14.sql - DE..lychitieu (sa (156)) \* SQLQuery13.sql - DE..lychitieu (sa (76)) \* DESKTOP-P37UKEM...u - dbo.TaiKhoan \* Quick Launch (Ctrl+Q)

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
=====
CREATE TRIGGER dbo.trg_DieuChinhSauCapNhatThu
ON dbo.QLThu
AFTER UPDATE
AS
BEGIN
    SET NOCOUNT ON;

    UPDATE TK
    SET TK.TongTien = TK.TongTien - D.SoTienThu + I.SoTienThu
    FROM dbo.TaiKhoan TK
    INNER JOIN inserted I ON TK.TenDangNhap = I.TenDangNhap
    INNER JOIN deleted D ON TK.TenDangNhap = D.TenDangNhap;

```

100 %

Messages

Command completed successfully.

Completion time: 2026-04-24T00:26:37.8118976+07:00

Query executed successfully.

DESKTOP-P37UKEM (16.0 RTM) sa (158) qlchitieu 00:00:00 0 rows

Ready

In 37 Col 3 Ch 3 INV

Sau chi:

DESKTOP-P37UKEM.qlchitieu - dbo.QLChi - Microsoft SQL Server Management Studio

File Edit View Project Query Designer Tools Help

Object Explorer

DESKTOP-P37UKEM...eu - dbo.TaiKhoan DESKTOP-P37UKEM...u - dbo.QLChi

idChi	NgayChi	SoTienChi	LyDoChi	TenDangN...
1	2025-04-23	234.00	egg	long
2	2025-04-23	435.00	egg	lien
**	NULL	NULL	NULL	NULL

DESKTOP-P37UKEM...eu - dbo.QLChi

Connect - qlchitieu

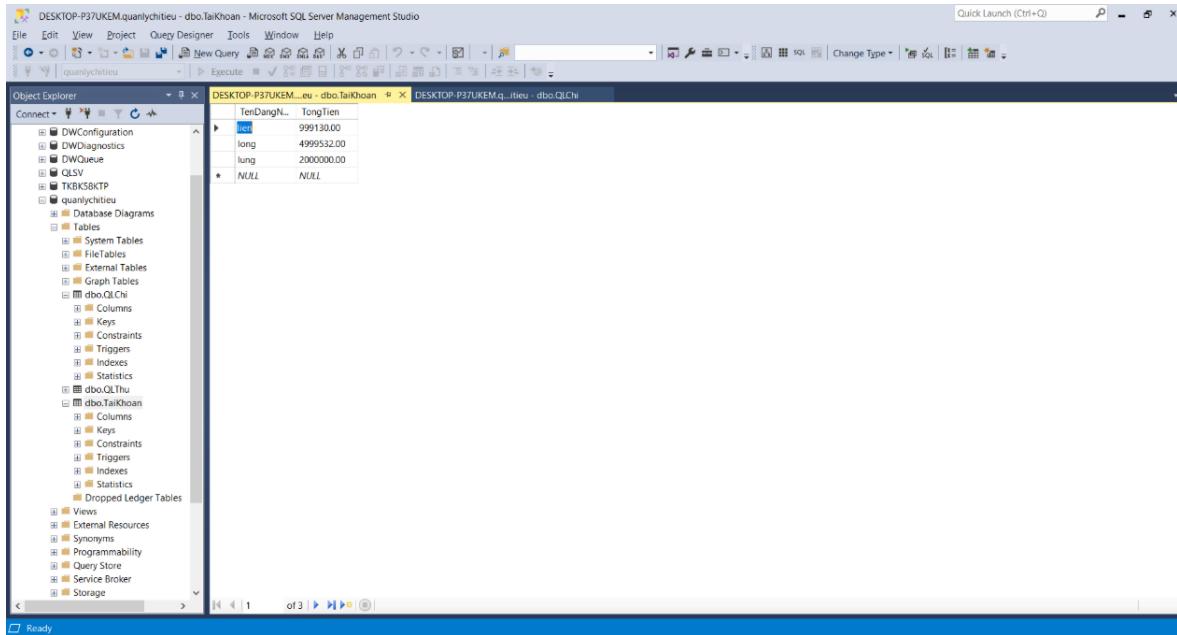
Databases Tables System Tables FileTables External Tables Graph Tables dbo.QLChi Columns Keys Constraints Triggers Indexes Statistics

dbo.QLThu Columns Keys Constraints Triggers Indexes Statistics

Views External Resources Synonyms Programmability Query Store Service Broker Storage

Ready

Kết quả:



**KẾT LUẬN:** Những Trigger trên đã giúp em hoàn thành bài tập được giao, ngăn chặn việc cố gắng chèn thêm những dữ liệu phi chuẩn vào bảng và tự động cập nhật dữ liệu 1 cách logic từ các bảng có liên kết khi dữ liệu của 1 hoặc nhiều trong các bảng bị thay đổi.

## 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.

---

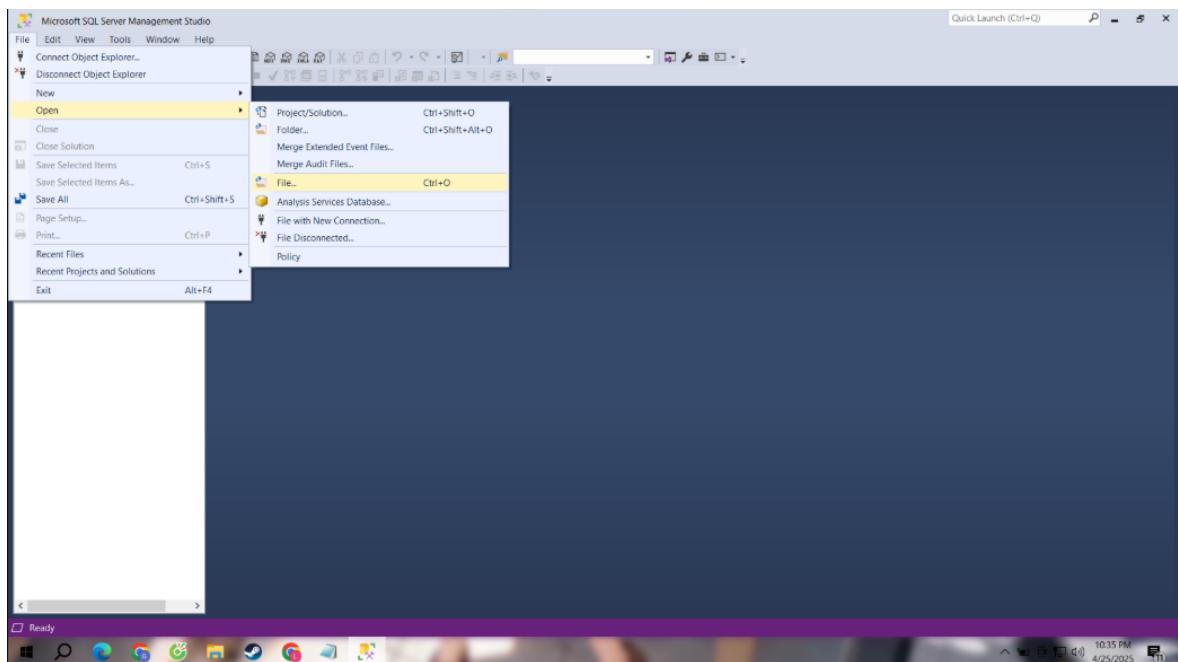
8. nhập sql để tìm xem có những sv nào có sđt sai khác chỉ 1 số so với sđt của em.

9. BẢNG SV CÓ HƠN 9000 ROWS, HÃY LIỆT KÊ TẤT CẢ CÁC SV NGÀNH KMT, SẮP XẾP THEO TÊN VÀ HỌ ĐỆM, KIÊU 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)

Bài Làm:

1. import dữ liệu:



Sau đó create database ( ở đây e đặt là sv\_tnut cho tiện với lệnh use sẵn có, ngoài ra có thể tùy biến):

Sau khi tạo database thì mở query mới, paste dữ liệu của file sv\_tnut.sql vừa mở rồi execute:

sv\_trut.sql - DESKTOP-P37UKEM sv\_trut (sa (160)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Quick Launch (Ctrl+Q) 

New Query                                          

Object Explorer                                            

Connect                                            

DESKTOP-P37UKEM (SQL Server 16.0.100)                                            

Object Explorer                                            

Connect                                            

Object Explorer                                            

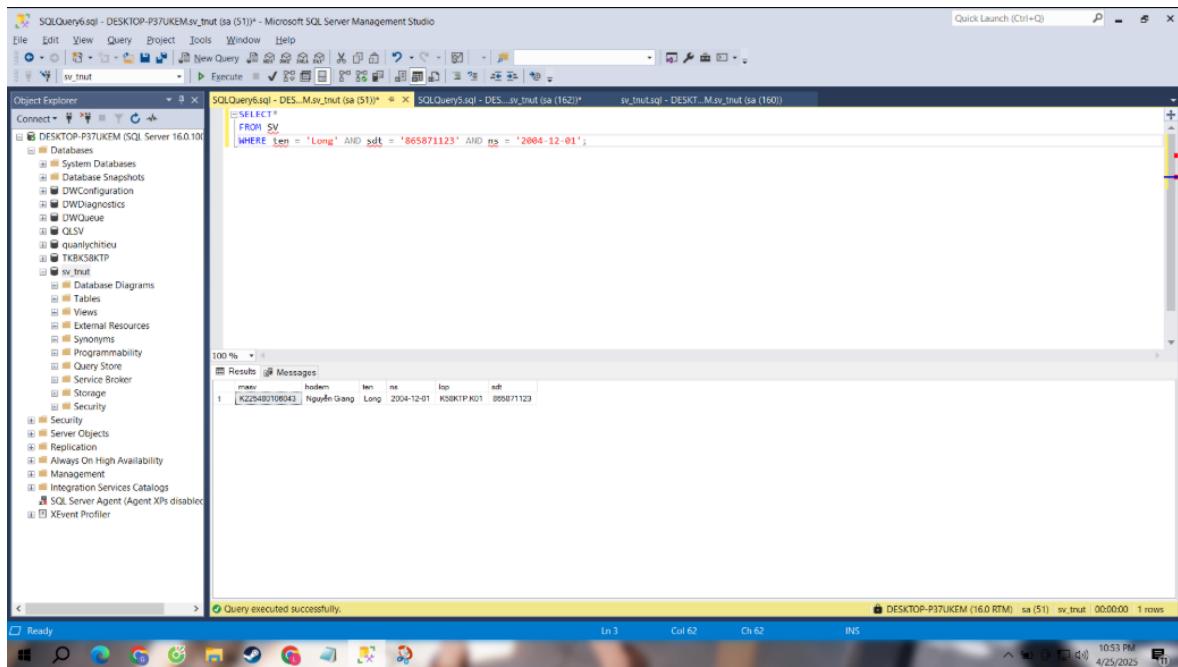
Object Explorer                                            

Object Explorer                                            

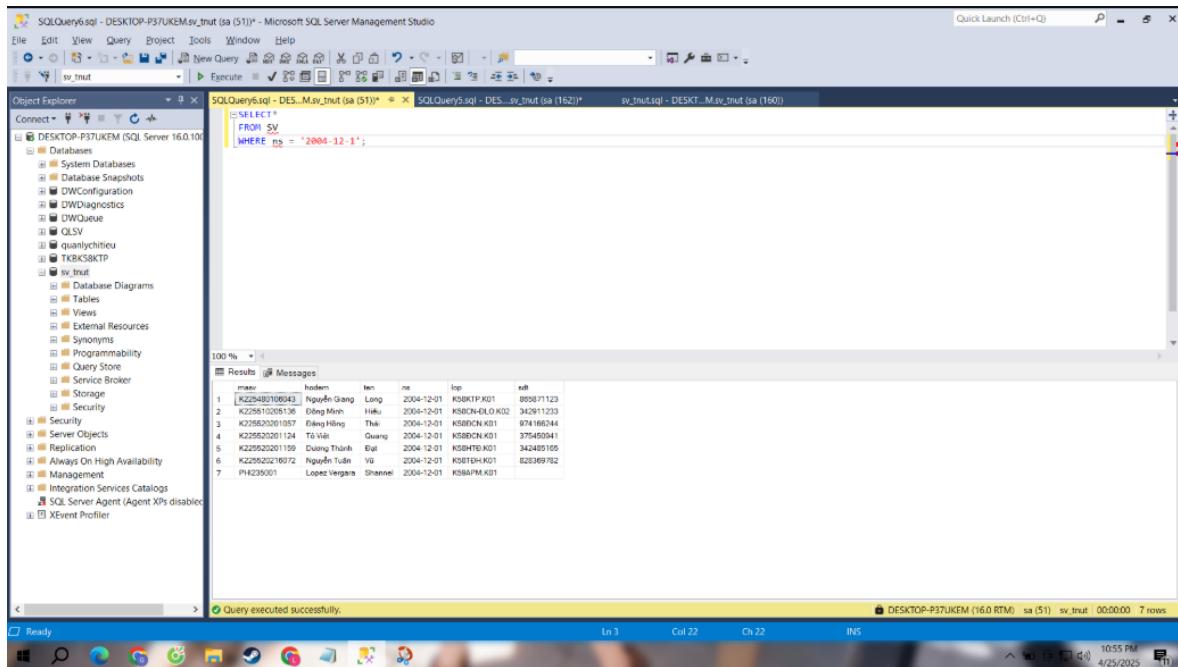
Object Explorer                                            

Object Explorer          <img alt

2. Truy vấn dữ liệu đầu vào: Để truy vấn e tạo 1 query mới, ở đây e sẽ dùng lệnh select như trên để ảnh đế thực hiện:



3.Nhập sql để tìm có sv nào trùng ngày tháng năm sinh với em Vẫn tiếp tục là lệnh select với cú pháp như ảnh trên:



4. nhập sql để tìm xem có những sv nào trùng ngày và tháng sinh với em? Vẫn tiếp tục là lệnh select với cú pháp như ảnh trên:

```

SELECT *
FROM SV
WHERE DAY(ns) = 01 AND MONTH(ns) = 12;

```

Results

maSV	hoten	ten	ns	lop	ktit
1	K1652020103270	Lưu Lú	Bán	1999-12-01	K520CM.02
2	K194905221002	LA Nguyễn Duy	Bík	2001-12-01	K55AP1
3	K205202016013	Lê Tiên	Thành	1999-12-01	K56TH.01
4	K205202016015	Đặng Hồi	Huy	2000-12-01	K56TH.07
5	K219202010377	Đặng Chí	Công	2003-12-01	K57DCM.03
6	K219202010378	Nguyễn Văn	Bắc	2003-12-01	K57DCM.03
7	K219202010319	Đặng Văn	Đông	2003-12-01	K57DCM.03
8	K219202114558	Nguyễn Minh	Quang	2000-12-01	K57DCM.03
9	K219202010113	Hoa Kỳ	Văn	2003-12-01	K57DCM.01
10	K219202010169	Đinh Quốc	Ánh	2003-12-01	K57DTH.01
11	K219202010174	Nguyễn Tài	Tuệ	2003-12-01	K57DTH.01
12	K225400108043	Nguyễn Giang	Long	2003-12-01	K58KTP.01
13	K22510205139	Đặng Minh	Hữu	2003-12-01	K58KCN-DL0.02
14	K22510205139	Đặng Hồng	Thái	2003-12-01	K58KCN.01
15	K225202001194	Đặng Văn	Quang	2003-12-01	K58KCN.01
16	K2252020201159	Đặng Thành	Đạt	2003-12-01	K58KTC.01
17	K225202016072	Nguyễn Tuân	Vũ	2004-12-01	K58KTH.01

Query executed successfully.

5.nhập sql để tìm xem có những sv nào trùng tháng và năm sinh với em?

```

SELECT *
FROM SV
WHERE MONTH(ns) = 12 AND YEAR(ns) = 2004;

```

Results

maSV	hoten	ten	ns	lop	ktit
1	1151228008	Lê Trần Minh	Hoàng	2004-12-03	LT23CN-KT0.01
2	11512401029	Tạ Quang	Duy	2004-12-15	BT23CN-KT0.01
3	B022341029	Nguyễn Văn	Mùa	2004-12-15	BT23THH.01
4	B022341023	Nguyễn Văn	Nhật	2004-12-18	BT23THH.01
5	CPC239023	Lan	Chí	2004-12-11	K59KDC.01
6	IDN23902	I Komang ary	Ananta	2004-12-27	K58AP1.01
7	K225202010116	Mai Thị Thành	Tâm	2004-12-27	K58NNK.01
8	K225400106002	Nguyễn Tuân	Ánh	2004-12-18	K58KTP.01
9	K225400106033	Đặng Anh	Khai	2004-12-07	K58KTP.01
10	K225400106043	Nguyễn Giang	Long	2004-12-01	K58KTP.01
11	K225400106044	Nguyễn Văn	Đông	2004-12-01	K58KTP.01
12	K226912010111	Hồ Văn	Đức	2004-12-12	K58KCN-GX1.01
13	K22691201021	Hồ Văn	Hồ	2004-12-10	K58KCN-GX1.01
14	K226912020011	Nguyễn Thành	Hải	2004-12-11	K58KCN-CTM.01
15	K226912020037	Trần Long	Vũ	2004-12-05	K58KCN-CTM.01
16	K226912020021	Trần Trung	Hiếu	2004-12-12	K58KCN-BLD.01
17	K22691202028	Nguyễn Quang	Huy	2004-12-18	K58KCN-GND.01

Query executed successfully.

6.nhập sql để tìm xem có những sv nào trùng tên với em?

```

SELECT *
FROM SV
WHERE ten = 'Long';

```

mess	hodem	ten	ns	lop
1	1191230551	Nguyễn Văn	Long	2002-09-12
2	11512465012	Hồng Thành	Long	2001-10-13
3	11512465311	Tâm Quý	Long	2002-12-18
4	11512465312	Thiên Phú	Long	2001-09-18
5	BN23235929	Nguyễn Hữu	Long	2001-01-04
6	BN23235930	Nguyễn Tiến	Long	2000-12-12
7	BN23235952	Nguyễn Thành	Long	2002-10-02
8	K165520120109	Nguyễn Đức	Long	1998-09-23
9	K185510301061	Cao Hoang	Long	2000-10-29
10	K185520114186	Đặng Hải	Long	2000-11-29
11	K185520114187	Đoàn Ngọc	Long	2000-11-29
12	K185520109191	Đỗ Văn Hùng	Long	2000-09-19
13	K198520210906	Nguyễn Đức	Long	2001-09-21
14	K205510205106	Huang	Long	2002-09-19
15	K205510205109	Nguyễn Thế	Long	2002-03-09
16	K205510301148	Trần Văn	Long	2002-05-21
17	K205510301149	Trịnh Hải	Long	2002-05-03

7.nhập sql để tìm xem có những sv nào trùng họ và tên đệm với em.

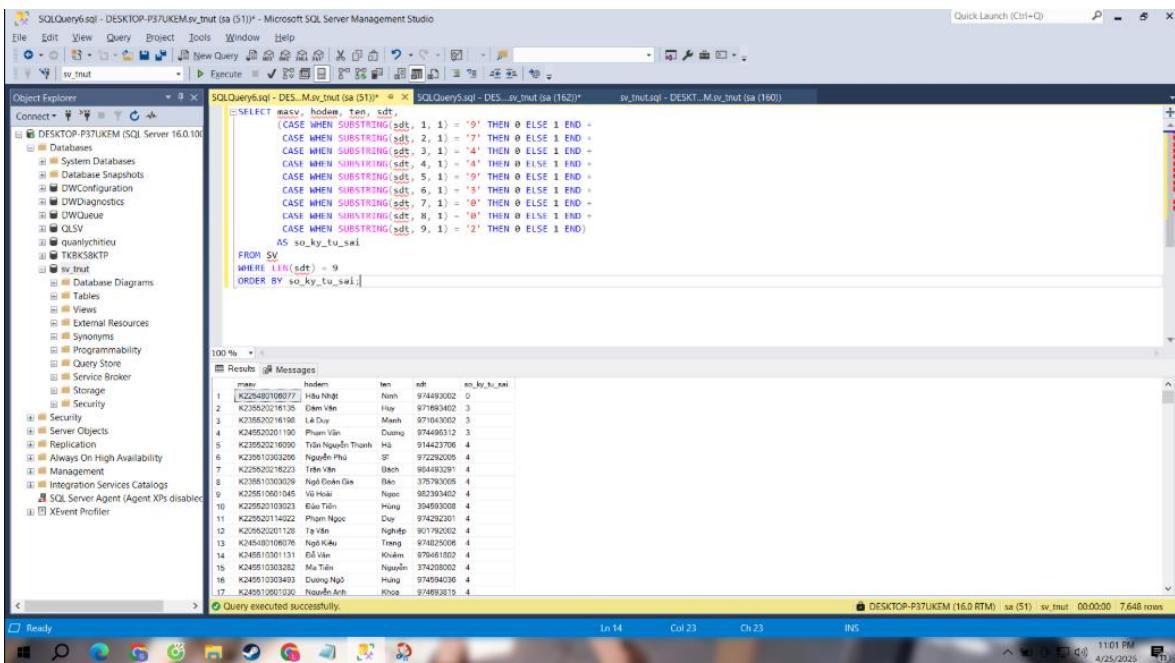
```

SELECT *
FROM SV
WHERE hodem = 'Nguyễn Giang';

```

mess	hodem	ten	ns	lop
1	K025620207070	Nguyễn Giang	Nam	2002-07-01
2	K255407106043	Nguyễn Giang	Long	2004-12-01
3	K258810301203	Nguyễn Giang	ßen	2004-05-01

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

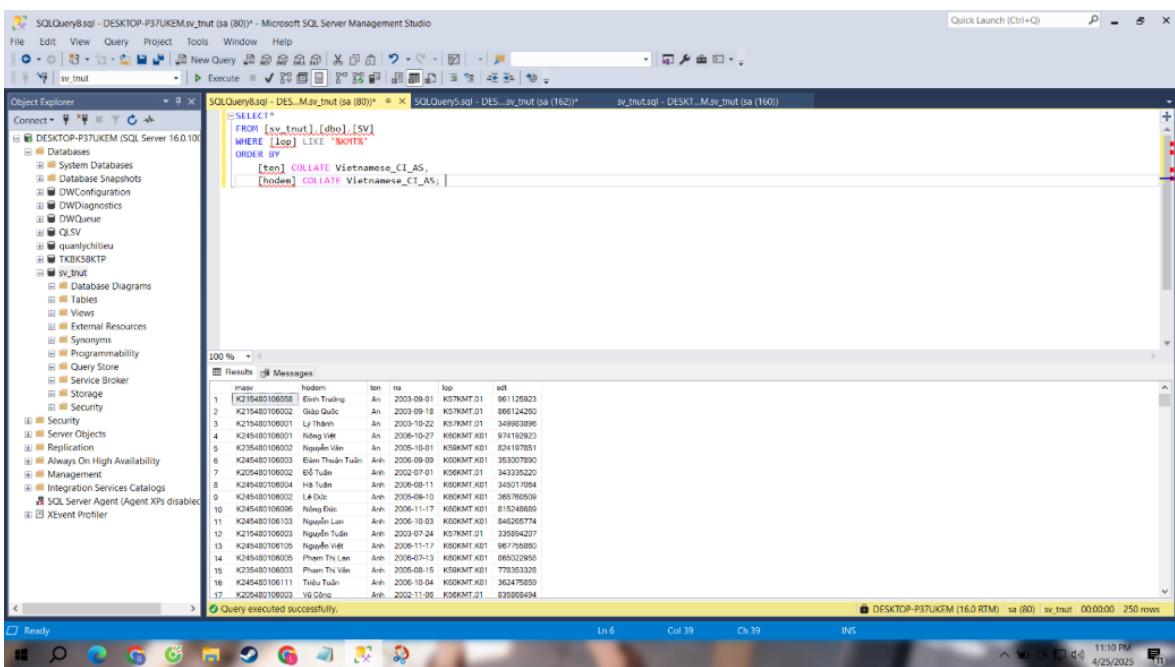


```

SELECT masv, hodem, ten, lop, sdt
FROM SV
WHERE lop LIKE '%KMT'
ORDER BY so_ky_tu_seu;

```

9.BẢNG SV CÓ HƠN 9000 ROWS, HÃY LIỆT KÊ TẤT CẢ CÁC SV NGÀNH KMT, SẮP XẾP THEO TÊN VÀ HỌ ĐỆM, KIỀU TIẾNG VIỆT, GIẢI THÍCH.



```

SELECT *
FROM [sv_trut].[dbo].[SV]
WHERE [ten] LIKE '%KMT'
ORDER BY
    [ten] COLLATE Vietnamese_CI_AS,
    [hodem] COLLATE Vietnamese_CI_AS;

```

where ...: lấy những dòng có chữ KMT order by: sắp xếp kết quả [ten] COLLATE Vietnamese\_CI\_AS: Sắp xếp theo tên sinh viên, dùng bộ mã chuẩn tiếng Việt để phân biệt [hodem] COLLATE Vietnamese\_CI\_AS: Nếu trùng tên thì sắp xếp tiếp theo họ đệm theo tiếng Việt.

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 VƯỚNG MẮC)?

```

SELECT *
FROM sv_trust
WHERE [ten] LIKE 'N%My%'  

    AND [ten] LIKE 'N%Linh%'  

    AND ( [ten] LIKE N'Anh' OR [ten] LIKE N'Trang' OR [ten] LIKE N'Lan' OR  

    [ten] LIKE N'Hoai' OR [ten] LIKE N'Huong' OR [ten] LIKE N'Linh' OR  

    [ten] LIKE N'Giang' OR [ten] LIKE N'Thuy' OR [ten] LIKE N'Trinh' OR  

    [ten] LIKE N'Hanh' OR [ten] LIKE N'Thao' OR [ten] LIKE N'Hang' OR  

    [ten] LIKE N'Van' OR [ten] LIKE N'My' OR [ten] LIKE N'Mai' OR  

    [ten] LIKE N'Ngoc' OR [ten] LIKE N'Yen' OR [ten] LIKE N'Dieu'
);

```

matru	hoDem	ten	ns	lop	xld	
1	K245480106110	Nguyễn Thị Chia	My	2005-04-29	K60KMT K01	37867751
2	K245480106020	Nguyễn Thị Uy	Lin	2005-08-15	K60KMT K01	77853328
3	K245480106043	Nguyễn Thị Ngọc	Linh	2005-07-13	K60KMT K01	81773386
4	K245480106026	Phạm Thị Lan	Anh	2006-07-13	K60KMT K01	86022908
5	K245480106077	Võ Thị Huân	Trang	2006-06-23	K60KMT K01	87767309
6	K245480106083	Phạm Thị Thảo	Vân	2006-09-01	K60KMT K01	980373542
7	K245480106102	Nguyễn Thị Mai	Hồng	2006-08-30	K60KMT K01	0344 379 588

Vì do kh có cột nào về giới tính nên em sẽ liệt kê những tên mà em có thể biết là của các bạn nữ( hoặc sử dụng dữ liệu trên mạng hoặc hỏi chat ai r cop), sau đó dùng lệnh Where và or những dữ liệu về tên cần thiết để lọc và liệt kê. Em nghĩ sẽ khó có thể liệt kê đầy đủ với cách trên nhưng mà phần trăm sai giới tính sẽ thấp.

---

Link Bài tập 1,2,3: <https://github.com/Shiwaratu2004/Nguyen-Long#>



Link Bài tập 4: <https://github.com/Shiwaratu2004/baitap4>



Link Bài tập 5: <https://github.com/Shiwaratu2004/baitap5>



Link Bài tập 6: <https://github.com/Shiwaratu2004/baitap6>



