**BÀI TẬP THỰC HÀNH MÔN**

**HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU SQL SERVER**

**Ngày:** 13/11/2023

**Họ và tên sinh viên:** Trần Nguyễn Gia Long **Lớp:** 10\_ĐH\_CNTT2

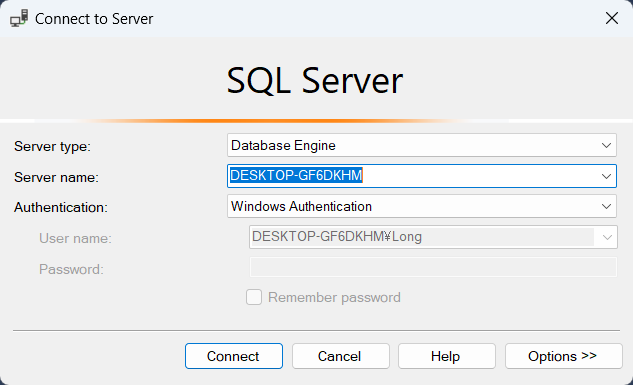
**MSSV:** 1050080059

# Module 1:

## Phần 1: SQL Server Management Studio

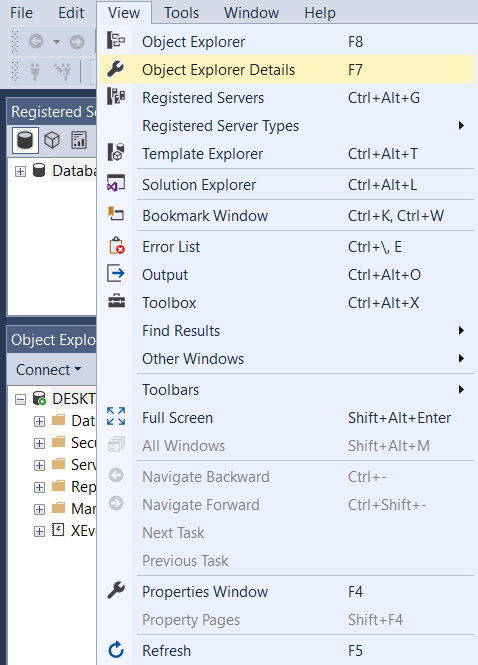
### 1. Khởi động SQL Server Management Studio

- Start🡪 Program File🡪 MicroSoft SQL Server 2008🡪SQL Server Management studio

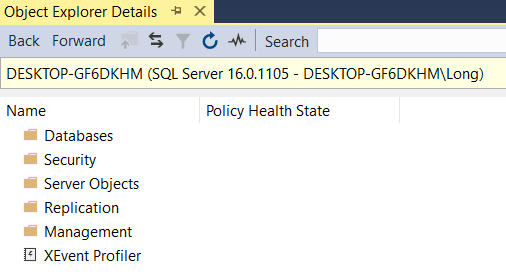


- Chọn Connect (Kết nối).

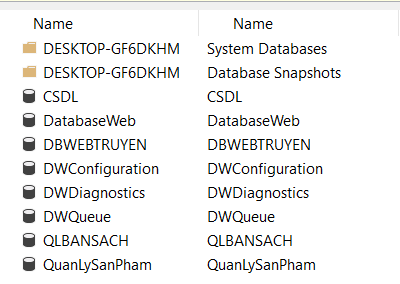
### 2. Vào menu View, Chọn Object Explorer Details



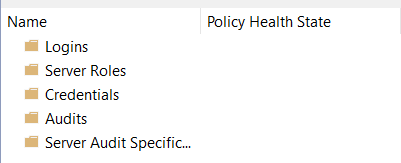
- Lần lượt mở các nhánh của cây MicroSoft SQL Servers.



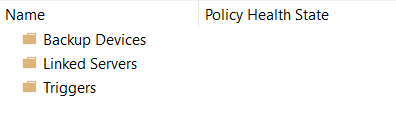
* Databases



* Security



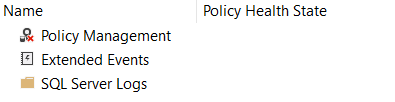
* Server Objects



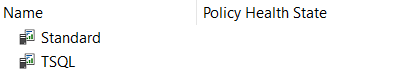
* Replication



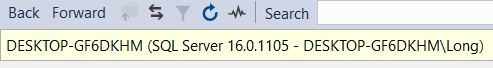
* Management



* XEvent Provider



- Sơ lược cửa sổ, trình đơn, thanh công cụ

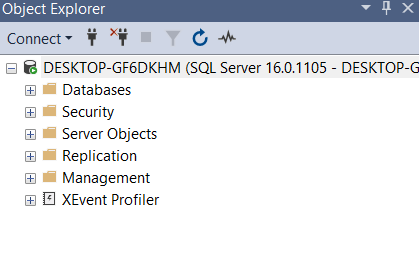


Gồm 7 thanh công cụ:

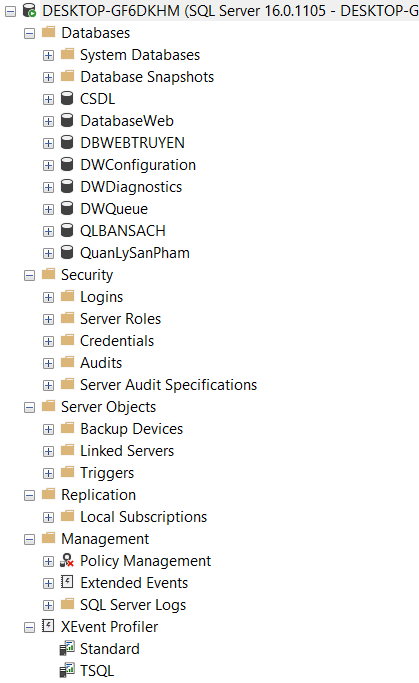
* Back: Di chuyển đến các mục đã hiển thị trước đó trong Object Explorer Details.
* Forward: Di chuyển đến màn hình tiếp theo sau khi chọn Back.
* Up: Di chuyển đến đối tượng hoặc thư mục cha.
* Synchronize: Đặt trọng tâm của Object Explorer vào đối tượng được chọn trong Object Explorer Details.
* Filter: Khi có sẵn, hiển thị một tập hợp con có thể cấu hình của các đối tượng.
* Refresh: Làm mới hiển thị trong Object Explorer Details.
* Search: Cung cấp một khu vực để nhập một thuật ngữ tìm kiếm cho một số đối tượng cơ sở dữ liệu.

### 3. Tại cửa sổ Object Explorer.

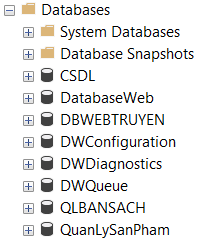
* Quan sát các thành phần đối tượng trên cửa sổ:
* Có 1 SQL Server Group. Tên Server là DESKTOP-GF6DKHM. Hiện tại đang Connect.



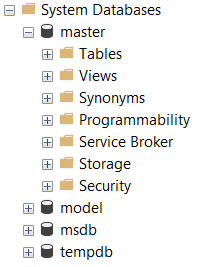
* Các thành phần trong Server hiện hành:



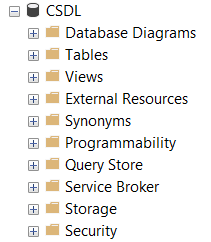
* Databases
* Security
* Server Objects
* Replication
* Management
* XEvent Provider
* Các database trong Server hiện hành:



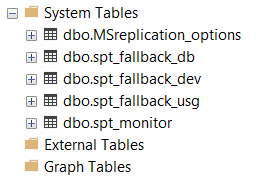
* System Database
* Database Snapshots
* CSDL
* DatabaseWeb
* DWConfiguration
* DWDiagnostics
* DWQueue
* QLBANSACH
* QuanLySanPham
* Trong mỗi Database có những đối tượng sau:
* Trong System Database, có 7 đối tượng gồm *Tables, View, Synonyms, Programmability, Service Broker, Storage, Security*. Các database khác như model, msdb, tempdb cũng tương tự.



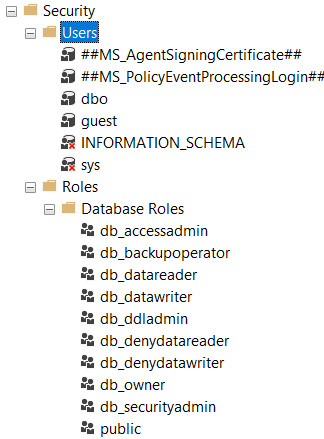
* Trong các Database còn lại, chỉ có thêm *Database Diagrams, External Resources, Query Store.*



* Mở database Master, khảo sát các đối tượng:
* Vào đối tượng Table, tìm hiểu cấu trúc và dữ liệu của bảng:



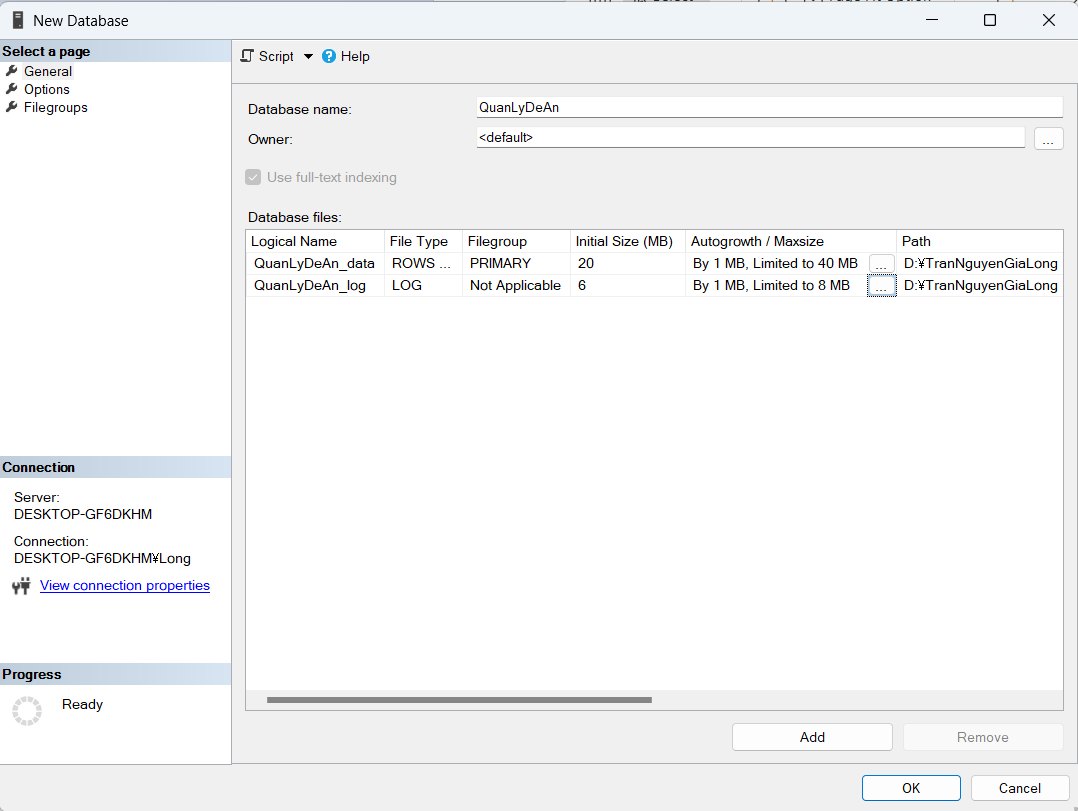
* Vào đối tượng Stored Procedures, tìm hiểu nội dung của các thủ tục sau:
* Vào đối tượng User, Role tìm hiểu nội dung của các thủ tục:



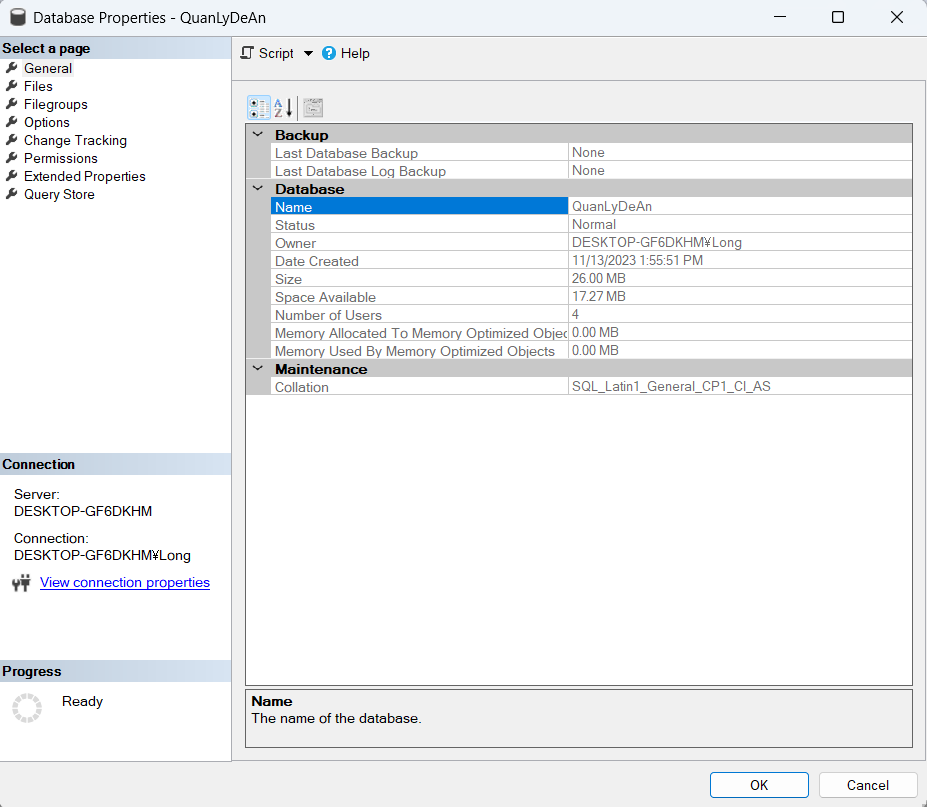
## Phần 2: Tạo và quản lý CSDL

### 1. Tạo CSDL từ menu

#### a. Click phải vào Database🡪 chọn New Database



#### b. Xem lại thuộc tính (properties) của CSDL QuanLyDeAn.

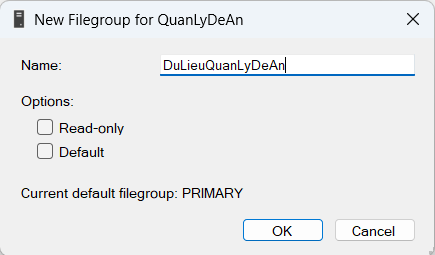


Thông tin thể hiện gồm:

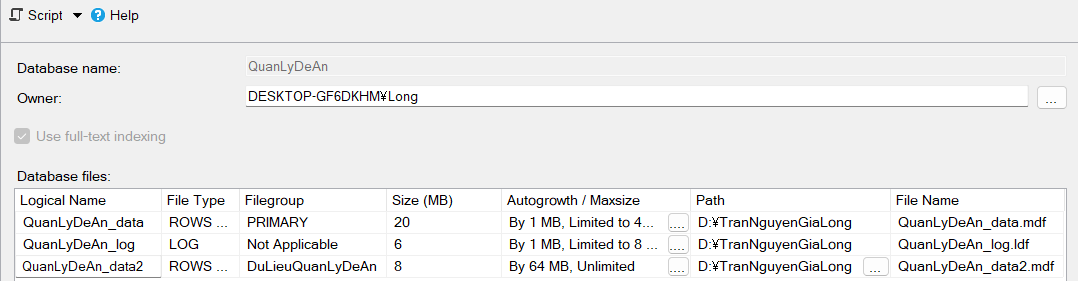
* **Tên cơ sở dữ liệu**: QuanLyDeAn
* **Trạng thái**: Normal
* **Chủ sở hữu**: DESKTOP-GF6DHM\Long
* **Ngày tạo**: 11/13/2023 1:55:51 PM
* **Kích thước**: 26.00 MB
* **Dung lượng trống**: 172.27 MB
* **Số người dùng**: 4
* **Bộ nhớ được cấp cho các đối tượng tối ưu hóa bộ nhớ:** 0.00 MB
* **Bảng mã:** SQL\_Latin1\_General\_CP1

#### c. Tại cửa sổ properties của CSDL, khai báo thêm

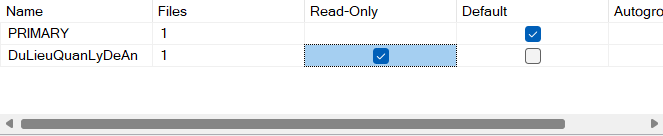
Tạo một filegroup mới có tên là DuLieuQuanLyDeAn



Một tập tin dữ liệu (data file) thứ hai nằm trong Group file vừa tạo ở trên và có thông số như sau Tên login của data file là QuanLyDeAn \_data2; Tên tập tin và đường dẫn vật lý của data file là D:\TranNguyenGiaLong\QuanLyDeAn\_data2.mdf.

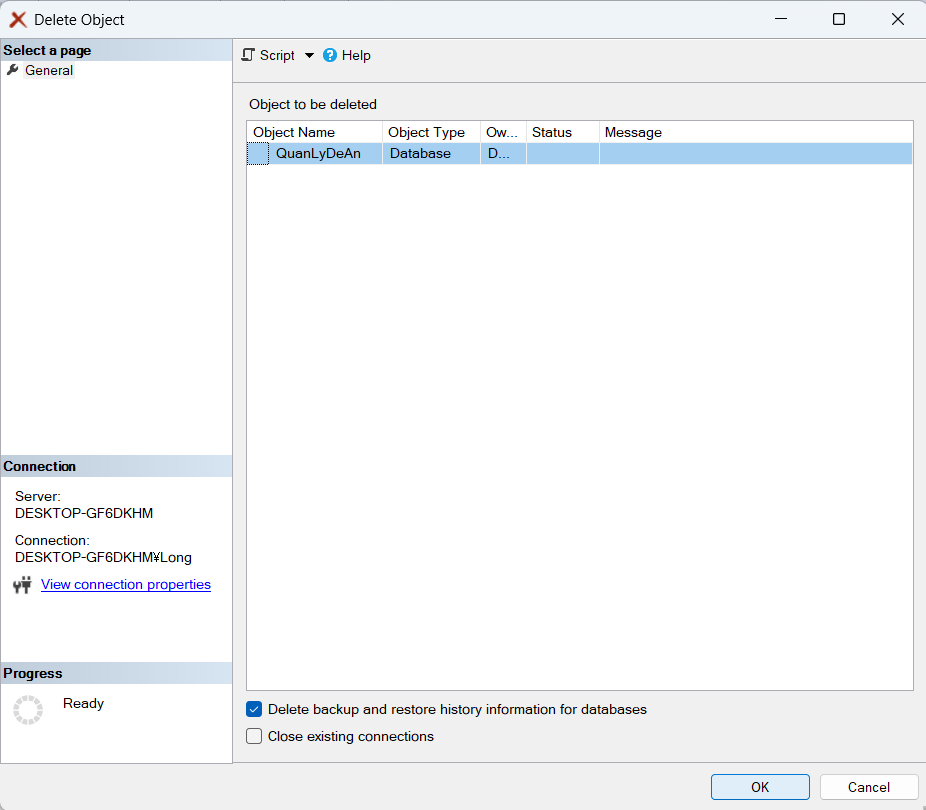


Chọn thuộc tính ReadOnly, sau đó đóng cửa sổ properies. Quan sát màu sắc của CSDL. Bỏ thuộc tính ReadOnly.



### 2. Thực hiện xóa CSDL quản lý thư viện vừa tạo và kiểm tra lại các files group đã bị xóa.

* Xoá CSDL Đề án.



### 3. Thực hiện tại cửa sổ Query Editor (lưu ý: sau mỗi lần có sự thay đổi thì phải dùng các lệnh để kiểm tra sự thay đổi đó) .

#### a. Dùng lệnh Create DataBase, tạo một CSDL với các tham số được liệt kê như trong bảng dưới. Lưu ý rằng CSDL này gồm một data file và nó được nằm trong primary filegroup

|  |  |
| --- | --- |
| THAM SỐ | GIÁ TRỊ |
| Database name | QuanLyDeAn |
| Tên logic của data file chính | QuanLyDeAn\_data1 |
| Tên tập tin và đường dẫn của data file chính | D:\TranNguyenGiaLong\QuanLyDeAn\_data1.mdf |
| Kích cỡ khởi tạo của CSDL | 10 MB |
| Kích cỡ tối đa của CSDL | 40 MB |
| Kích thước gia tăng tập tin CSDL | 1 MB |
| Tên logic của transaction log | QuanLyDeAn\_log |
| Tên tập tin và đường dẫn của transaction log | D:\TranNguyenGiaLong\QuanLyDeAn\_log.ldf |
| Kích cỡ khởi tạo của transaction log | 6 MB |
| Kích cỡ tối đa của transaction log | 8MB |
| Gia số gia tăng của transaction log | 1 MB |

CREATE DATABASE QuanLyDeAn

ON PRIMARY

(NAME = QuanLyDeAn\_data1,

FILENAME = 'D:\TranNguyenGiaLong\QuanLyDeAn\_data1.mdf',

SIZE = 10MB,

MAXSIZE = 40MB,

FILEGROWTH = 1MB)

LOG ON

(NAME = QuanLyDeAn\_Log,

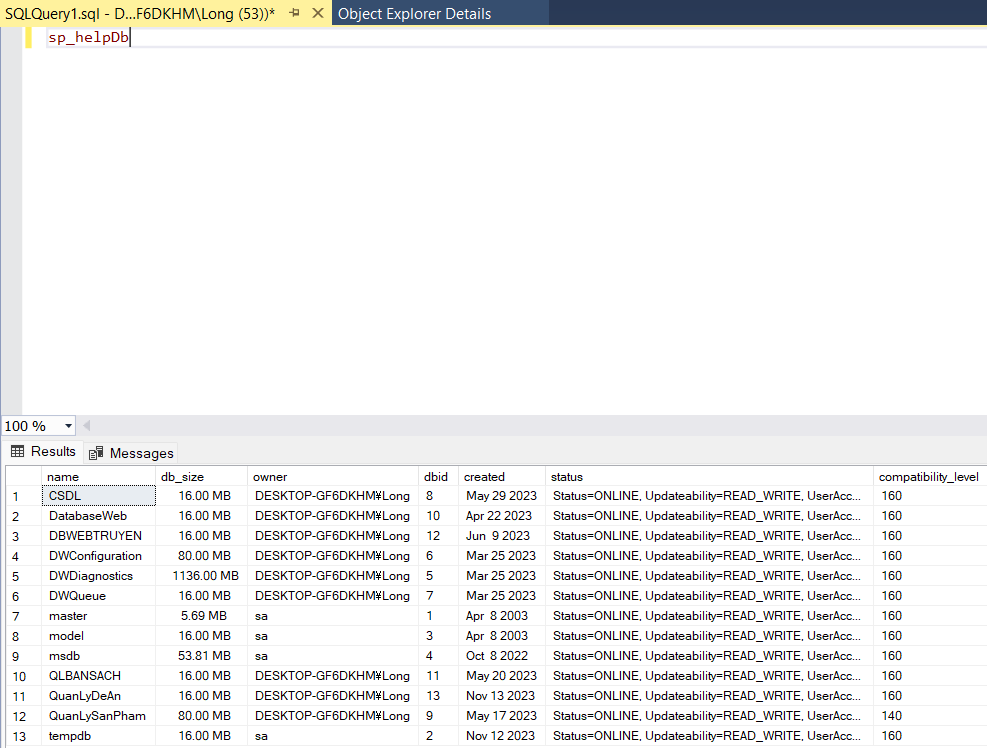
FILENAME = 'D:\TranNguyenGiaLong\QuanLyDeAn\_log.ldf',

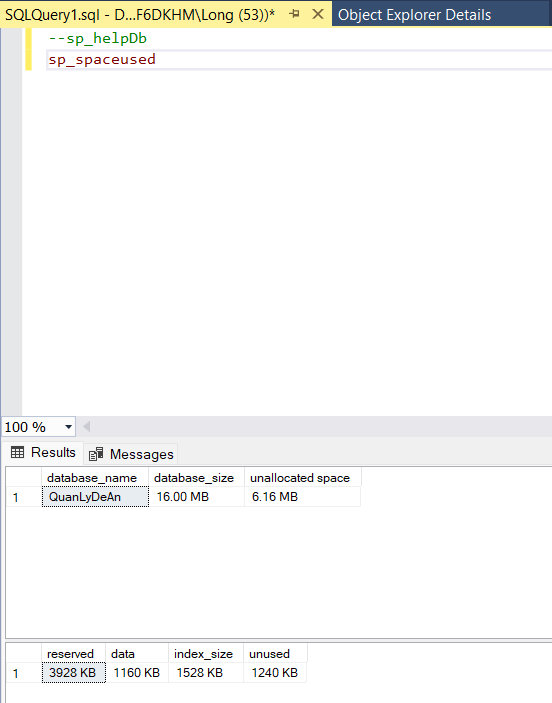
SIZE = 6MB,

MAXSIZE = 8MB,

FILEGROWTH = 1MB);

#### b. Xem lại thuộc tính của CSDL QuanLyDeAn bằng Management Studio và bằng thủ tục hệ thống sp\_helpDb, sp\_spaceused.



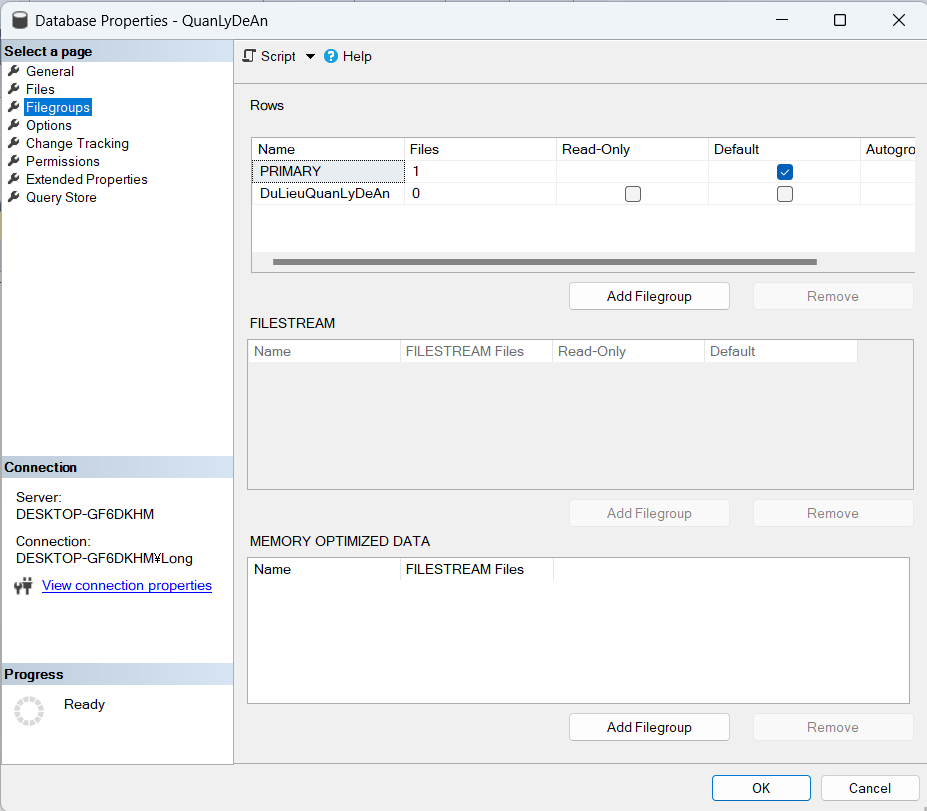


#### c. Thêm một filegroup có tên là DuLieu QuanLyDeAn (HD: dùng lệnh Alter DataBase <Tên Database> ADD FILEGROUP <Tên filegroup>)

ALTER DATABASE QuanLyDeAn

ADD FILEGROUP DuLieuQuanLyDeAn;

Kiểm tra filegroup đã tạo hay chưa.



#### d. Khai báo một secondary file có tên logic là QuanLyDeAn \_data2, tên vật lý QuanLyDeAn \_data2.ndf nằm ở T:\HoTenSV, các thông số khác tùy bạn chọn, data file này nằm trong file group là QuanLyDeAn. (HD: Dùng lệnh Alter Database …. ADD FILE …. TO FILEGROUP …)

ALTER DATABASE QuanLyDeAn

ADD FILE

(

NAME = QuanLyDeAn\_data2,

FILENAME = 'D:\TranNguyenGiaLong\QuanLyDeAn\_data2.ndf',

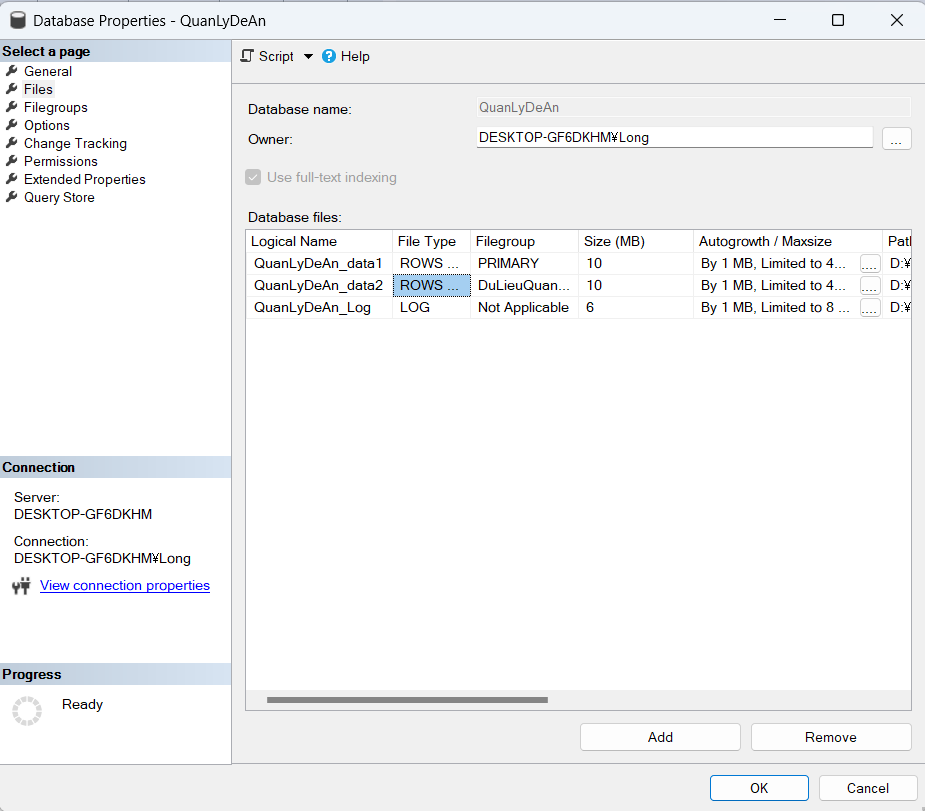
SIZE = 10MB,

MAXSIZE = 40MB,

FILEGROWTH = 1MB

)

TO FILEGROUP DuLieuQuanLyDeAn;



#### e. Dùng Books Online, bạn tìm hiểu thủ tục hệ thống sp\_helpfilegroup dùng để làm gì?

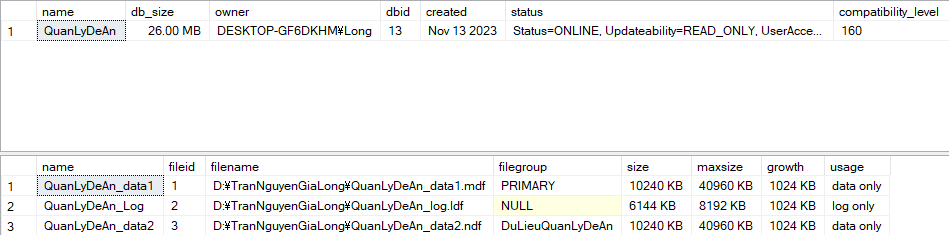
Thủ tục hệ thống sp\_helpfilegroup trong SQL Server được sử dụng để trả về các tên và thuộc tính của các filegroup liên quan đến cơ sở dữ liệu hiện tại

Trong đó, [ @filegroupname = ] 'name' là tên logic của bất kỳ filegroup nào trong cơ sở dữ liệu hiện tạiname là kiểu sysname, với giá trị mặc định là NULL. Nếu name không được chỉ định, tất cả các filegroup trong cơ sở dữ liệu hiện tại sẽ được liệt kê.

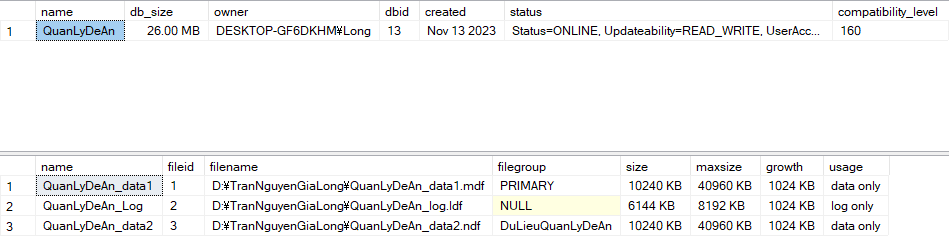
#### f. Dùng lệnh Alter Database … Set … để cấu hình cho CSDL QuanLyDeAn có thuộc tính là Read\_Only. Dùng sp\_helpDB để xem lại thuộc tính của CSDL. Hủy bỏ thuộc tính Read\_Only.

ALTER DATABASE QuanLyDeAn SET READ\_ONLY;

EXEC sp\_helpdb 'QuanLyDeAn';



ALTER DATABASE QuanLyDeAn SET READ\_WRITE;



#### g. Dùng lệnh Alter DataBase … MODIFY FILE … để tăng SIZE của QuanLyDeAn\_data1 thành 50 MB. Tương tự tăng SIZE của tập tin QuanLyDeAn \_log thành 10 MB.

ALTER DATABASE QuanLyDeAn

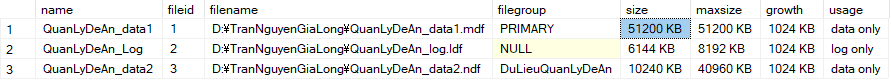
MODIFY FILE

(

NAME = QuanLyDeAn\_data1,

SIZE = 50MB

);



ALTER DATABASE QuanLyDeAn

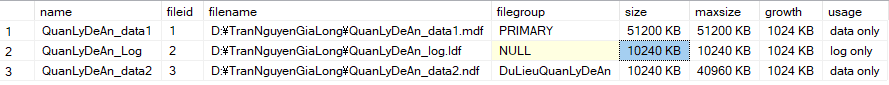
MODIFY FILE

(

NAME = QuanLyDeAn\_log,

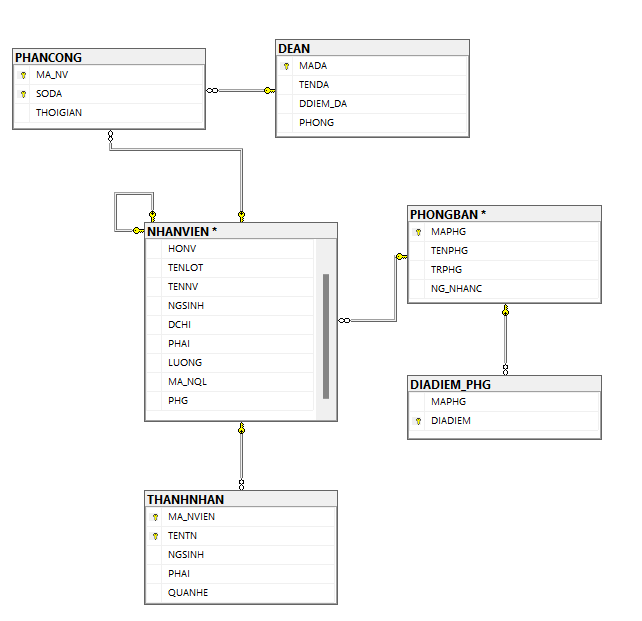
SIZE = 10MB

);



# Module 2

## 1. Cho CSDL Quản lý đề án, hãy tạo CSDL có tên\_QuanLyDeAn\_TenSV trong thư mục tùy ý. Sau đó, thiết kế cấu trúc các bảng có khóa chính khóa ngoại và kiểu dữ liệu theo lược đồ sau (thực hiện bằng DDL).



CREATE TABLE NHANVIEN (

MANV VARCHAR(9) NOT NULL,

HONV NVARCHAR(15) NULL,

TENLOT NVARCHAR(30) NULL,

TENNV NVARCHAR(30) NULL,

NGSINH SMALLDATETIME NULL,

DCHI NVARCHAR(150) NULL,

PHAI NVARCHAR(3) NULL,

LUONG NUMERIC(18,0) NULL,

MA\_NQL VARCHAR(9) NULL,

PHG VARCHAR(2) NULL

PRIMARY KEY (MANV)

)

CREATE TABLE DEAN(

MADA VARCHAR(2) NOT NULL,

TENDA NVARCHAR(50) NULL,

DDIEM\_DA VARCHAR(20) NULL,

PHONG VARCHAR(2) NULL

PRIMARY KEY (MADA)

)

CREATE TABLE PHANCONG (

MA\_NV VARCHAR(9) NOT NULL,

SODA VARCHAR(2) NOT NULL,

THOIGIAN NUMERIC(18,0) NULL

PRIMARY KEY (MA\_NV, SODA)

)

CREATE TABLE PHONGBAN(

MAPHG VARCHAR(2) NOT NULL,

TENPHG NVARCHAR(20) NULL,

TRPHG VARCHAR(9) NULL,

NG\_NHANCHUC SMALLDATETIME NULL

PRIMARY KEY (MAPHG)

)

CREATE TABLE DIADIEM\_PHG(

MAPHG VARCHAR(2) NOT NULL,

DIADIEM VARCHAR(20) NOT NULL,

PRIMARY KEY (DIADIEM),

FOREIGN KEY (MAPHG) REFERENCES PHONGBAN (MAPHG)

)

CREATE TABLE THANNHAN(

MA\_NVIEN VARCHAR(9) NOT NULL,

TENTN VARCHAR(20) NOT NULL,

NGSINH SMALLDATETIME NULL,

PHAI VARCHAR(3) NULL,

QUANHE VARCHAR(15) NULL,

PRIMARY KEY (TENTN, MA\_NVIEN),

)

ALTER TABLE PHANCONG WITH CHECK ADD CONSTRAINT [PhancongNhanvien] FOREIGN KEY([MA\_NV])

REFERENCES NHANVIEN ([MANV])

ALTER TABLE PHANCONG WITH CHECK ADD CONSTRAINT [PhancongThanh] FOREIGN KEY([SODA])

REFERENCES DEAN ([MADA])

ALTER TABLE THANHNHAN WITH CHECK ADD CONSTRAINT [TroThanh] FOREIGN KEY([MA\_NVIEN])

## REFERENCES NHANVIEN ([MANV])

## 2. Tìm hiểu về kiểu dữ liệu (datatype):

### a. Tìm hiểu và trả lời các câu hỏi sau:

* Có 4 loại datatype

Số chính xác (Exact numeric): bit, tinyint, smallint, int, bigint, decimal, numeric, money và smallmoney.

Số xấp xỉ (Approximate numeric): float và real.

Ngày và thời gian (Date and time): date, datetime, datetime2, datetimeoffset, smalldatetime, time.

Chuỗi ký tự (Character strings): char, varchar, text.

* Các kiểu dữ liệu hệ thống (system datatype) trong SQL Server được lưu trữ trong bảng sys.types
* Các kiểu dữ liệu do người dùng định nghĩa (User-defined datatypes) trong SQL Server được lưu trữ trong bảng sys.types

#### b. Vào màn hình Query Editor, chọn QuanLyDeAn là CSDL hiện hành, định nghĩa các datatype:

|  |  |
| --- | --- |
| **KIỂU DỮ LIỆU (DATA**  **TYPE)** | **MÔ TẢ DỮ LIỆU (DESCRIPTION OF**  **DATA)** |
| ISBN | 25 ký tự |
| SoDienThoai | 13 ký tự , chấp nhận NULL |
| Shortstring | Số ký tự thay đổi đến 15 ký tự |

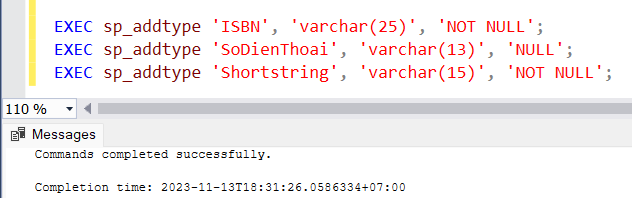
**Dùng thủ tục sp\_addtype hoặc Create Type để định nghĩa**

**Ví dụ: EXEC sp\_addtype SODienThoai, 'char(13)', NULL**

EXEC sp\_addtype 'ISBN', 'varchar(25)', 'NOT NULL';

EXEC sp\_addtype 'SoDienThoai', 'varchar(13)', 'NULL';

EXEC sp\_addtype 'Shortstring', 'varchar(15)', 'NOT NULL';

****

### c. Các User-defined datatype vừa định nghĩa được lưu trữ ở đâu và phạm vi sử dụng của nó ở đâu (trong toàn bộ một instance hay chỉ ở trong CSDL hiện hành).

Các kiểu dữ liệu do người dùng định nghĩa (User-defined datatypes) trong SQL Server được lưu trữ trong bảng sys.types

### d. Có bao nhiêu cách liệt kê danh sách các các User-Defined datatype vừa định nghĩa.

* Sử dụng INFORMATION\_SCHEMA.DOMAINS và sử dụng bảng sys.types

SELECT domain\_name, data\_type, character\_maximum\_length

FROM information\_schema.domains

ORDER BY domain\_name

Hoặc SELECT \* From Systype

### e. Muốn User-Defined datatype được dùng trong tất cả các CSDL thì bạn định nghĩa nó ở đâu?

Để một kiểu dữ liệu do người dùng định nghĩa (User-Defined datatype) có thể được sử dụng trong tất cả các cơ sở dữ liệu, cần định nghĩa nó ở mức server chứ không phải ở mức cơ sở dữ liệu hoặc bằng cách sử dụng câu lệnh SQL CREATE TYPE

### f. Bạn hãy tạo 1 bảng có tên là Nhanvien\_Backup(MaNV, Hoten, NgaySinh, Phai,DienThoai, ThanhPho) trong CSDL QuanLyDeAn và sử dụng User-defined datatype vừa định nghĩa ở trên. Trong đó, thêm các ràng buộc sau:

USE QuanLyDeAn;

CREATE TABLE Nhanvien\_Backup (

MaNV int PRIMARY KEY CHECK (MaNV >= 1 AND MaNV <= 1000),

HoTen nvarchar(50) NOT NULL,

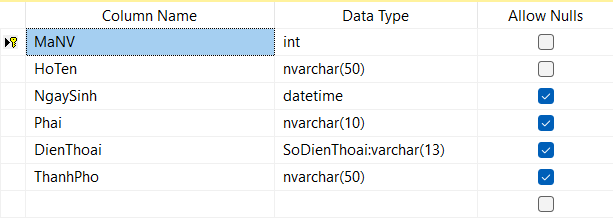
NgaySinh datetime,

Phai nvarchar(10),

DienThoai SoDienThoai UNIQUE,

ThanhPho nvarchar(50) DEFAULT 'Hồ Chí Minh'

);



### g. Hãy xóa kiểu dữ liệu SoDienThoai. Có xóa được hay không? Tại sao?

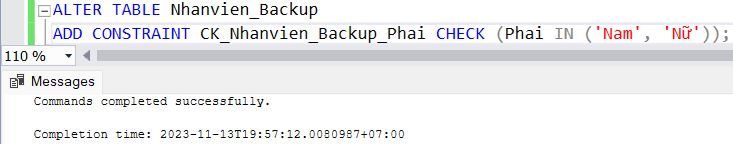
Xóa được vì đó là kiểu dữ liệu người dùng từ định nghĩa

3. Cài đặt thêm các ràng buộc check, rule, default:

### a. Hãy tạo ràng buộc cho cột Phai của bảng Nhanvien\_Backup chỉ có “Nam” hoặc “Nữ”

ALTER TABLE Nhanvien\_Backup

ADD CONSTRAINT CK\_Nhanvien\_Backup\_Phai CHECK (Phai IN ('Nam', 'Nữ'));



### b. Hãy tạo ràng buộc cho dữ liệu cho cột NgaySinh <=Ngayhientai-18 (năm) trong bảng Nhanvien\_Backup.

ALTER TABLE Nhanvien\_Backup

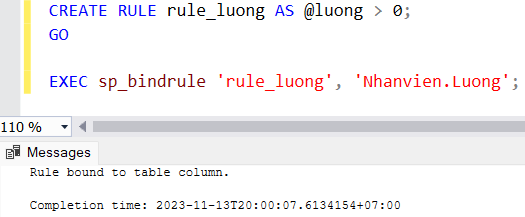
ADD CONSTRAINT CK\_Nhanvien\_Backup\_NgaySinh CHECK (NgaySinh <= DATEADD(year, -18, GETDATE()));

### c. Tạo 1 rule có tên là rule\_luong cho phép chứa giá trị lớn hơn 0, rồi gắn nó vào cột Luong trong bảng Nhanvien.

CREATE RULE rule\_luong AS @luong > 0;

GO

EXEC sp\_bindrule 'rule\_luong', 'Nhanvien.Luong';

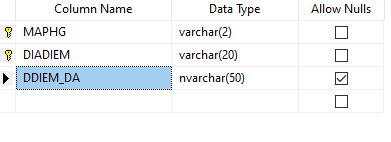


### d. Tạo 1 default có giá trị là ‘Tp.Hồ Chí Minh’ , sau đó gắn vào cột DDIEM\_DA của bảng DIADIEM.

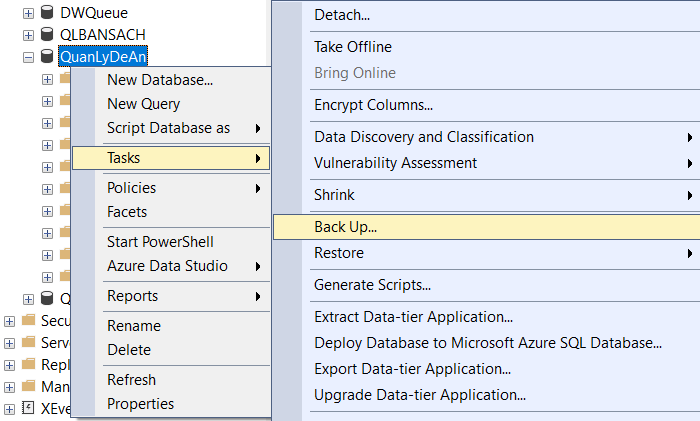
CREATE DEFAULT Default\_DDIEM\_DA AS 'Tp.Hồ Chí Minh';

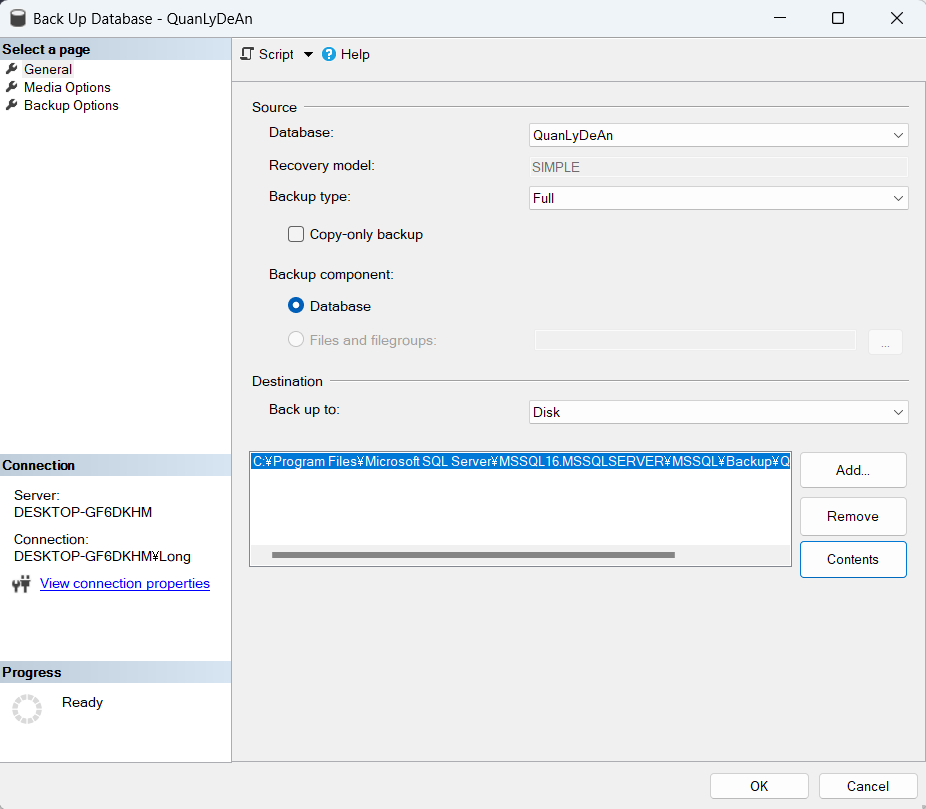
GO

EXEC sp\_bindefault 'Default\_DDIEM\_DA', 'DIADIEM.DDIEM\_DA';



## 4. Hãy thực hiện việc Backup và Retore CSDL Quanlydean.





# Module 3

## Phần 1: câu lệnh INSERT

### 1. Thực hiện việc attach để attach CSDL QuanLyDeAn được GV cung cấp tại phòng thực hành. Sau đó, chèn dữ liệu vào các bảng theo mẫu lệnh sau. Kiểm tra dữ liệu sau mỗi lần thực hiện.

BEGIN /\*\* NHANVIEN \*\*/

ALTER TABLE NHANVIEN

NOCHECK CONSTRAINT ALL

INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, NGSINH, DCHI, PHAI, LUONG, MA\_NQL, PHG)

VALUES (N'Đinh', N'Bá', N'Tiến', '009', '02/11/1960', N'119, Cống Quỳnh, TP.HCM', N'Nam', 30000, '005', 5)

INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, NGSINH, DCHI, PHAI, LUONG, MA\_NQL, PHG)

VALUES (N'Nguyễn', N'Thanh', N'Tùng', '005', '08/20/1962', N'222, Nguyễn Văn Cừ, TP.HCM', N'Nam', 40000, '006', 5)

INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, NGSINH, DCHI, PHAI, LUONG, MA\_NQL, PHG)

VALUES (N'Bùi', N'Ngọc', N'Hằng', '007', '03/11/1954', N'332, Nguyễn Thái Học, TP.HCM', N'Nam', 25000, '001', 4)

INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, NGSINH, DCHI, PHAI, LUONG, MA\_NQL, PHG)

VALUES (N'Lê', N'Quỳnh', N'Như', '001', '02/01/1967', N'291, Hồ Văn Huê, TP.HCM', N'Nữ', 43000, '006', 4)

INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, NGSINH, DCHI, PHAI, LUONG, MA\_NQL, PHG)

VALUES (N'Nguyễn', N'Mạnh', N'Hùng', '004', '03/04/1967', N'95, Bà Rịa - Vũng Tàu', N'Nam', 38000, '005', 5)

INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, NGSINH, DCHI, PHAI, LUONG, MA\_NQL, PHG)

VALUES (N'Trần', N'Thanh', N'Tâm', '003', '05/04/1957', N'34, Mai Thị Lự, TP.HCM', N'Nam', 25000, '005', 5)

INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, NGSINH, DCHI, PHAI, LUONG, MA\_NQL, PHG)

VALUES (N'Trần', N'Hồng', N'Quang', '008', '09/01/1967', N'45, Lê Hồng Phong, TP.HCM', N'Nam', 25000, '001', 4)

INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, NGSINH, DCHI, PHAI, LUONG, PHG)

VALUES (N'Phạm', N'Văn', N'Vinh', '006', '01/01/1965', N'45, Trưng Vương', N'Nữ', 55000, 1)

ALTER TABLE NHANVIEN

CHECK CONSTRAINT ALL

END



BEGIN /\*\* PHONGBAN \*\*/

ALTER TABLE PHONGBAN

NOCHECK CONSTRAINT ALL

INSERT INTO PHONGBAN (TENPHG, MAPHG, TRPHG, NG\_NHANCHUC)

VALUES (N'Nghiên cứu', 5, '005', '05/22/1978')

INSERT INTO PHONGBAN (TENPHG, MAPHG, TRPHG, NG\_NHANCHUC)

VALUES (N'Điều hành', 4, '008', '01/01/1985')

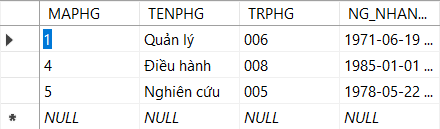
INSERT INTO PHONGBAN (TENPHG, MAPHG, TRPHG, NG\_NHANCHUC)

VALUES (N'Quản lý', 1, '006', '06/19/1971')

ALTER TABLE PHONGBAN

CHECK CONSTRAINT ALL

END



BEGIN /\*\* DEAN \*\*/

ALTER TABLE DEAN

NOCHECK CONSTRAINT ALL

INSERT INTO DEAN (TENDA, MADA, DDIEM\_DA, PHONG)

VALUES (N'Sản phẩm X', 1, N'Vũng Tàu', 5)

INSERT INTO DEAN (TENDA, MADA, DDIEM\_DA, PHONG)

VALUES (N'Sản phẩm Y', 2, N'Nha Trang', 5)

INSERT INTO DEAN (TENDA, MADA, DDIEM\_DA, PHONG)

VALUES (N'Sản phẩm Z', 3, N'TP.HCM', 5)

INSERT INTO DEAN (TENDA, MADA, DDIEM\_DA, PHONG)

VALUES (N'Tin học hóa', 10, N'Hà Nội', 4)

INSERT INTO DEAN (TENDA, MADA, DDIEM\_DA, PHONG)

VALUES (N'Cáp quang', 20, N'TP.HCM', 1)

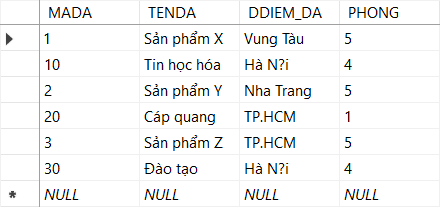
INSERT INTO DEAN (TENDA, MADA, DDIEM\_DA, PHONG)

VALUES (N'Đào tạo', 30, N'Hà Nội', 4)

ALTER TABLE DEAN

CHECK CONSTRAINT ALL

END



BEGIN /\*\* THANNHAN \*\*/

ALTER TABLE THANNHAN

NOCHECK CONSTRAINT ALL

INSERT INTO THANNHAN (MA\_NVIEN, TENTN, PHAI, NGSINH, QUANHE)

VALUES ('005', N'Trinh', N'Nữ', '04/05/1976', N'Con gái')

INSERT INTO THANNHAN (MA\_NVIEN, TENTN, PHAI, NGSINH, QUANHE)

VALUES ('005', N'Khang', N'Nam', '10/25/1973', N'Con trai')

INSERT INTO THANNHAN (MA\_NVIEN, TENTN, PHAI, NGSINH, QUANHE)

VALUES ('005', N'Phương', N'Nữ', '05/03/1948', N'Vợ chồng')

INSERT INTO THANNHAN (MA\_NVIEN, TENTN, PHAI, NGSINH, QUANHE)

VALUES ('001', N'Minh', N'Nam', '02/29/1932', N'Vợ chồng')

INSERT INTO THANNHAN (MA\_NVIEN, TENTN, PHAI, NGSINH, QUANHE)

VALUES ('009', N'Tiến', N'Nam', '01/01/1978', N'Con trai')

INSERT INTO THANNHAN (MA\_NVIEN, TENTN, PHAI, NGSINH, QUANHE)

VALUES ('009', N'Châu', N'Nữ', '12/30/1978', N'Con gái')

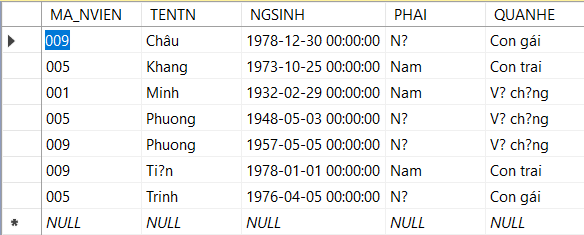
INSERT INTO THANNHAN (MA\_NVIEN, TENTN, PHAI, NGSINH, QUANHE)

VALUES ('009', N'Phương', N'Nữ', '05/05/1957', N'Vợ chồng')

ALTER TABLE THANNHAN

CHECK CONSTRAINT ALL

END



BEGIN /\*\* DIADIEM\_PHG \*\*/

ALTER TABLE DIADIEM\_PHG

NOCHECK CONSTRAINT ALL

INSERT INTO DIADIEM\_PHG (MAPHG, DIADIEM)

VALUES (1, N'TP.HCM')

INSERT INTO DIADIEM\_PHG (MAPHG, DIADIEM)

VALUES (4, N'Hà Nội')

INSERT INTO DIADIEM\_PHG (MAPHG, DIADIEM)

VALUES (5, N'Vũng Tàu')

INSERT INTO DIADIEM\_PHG (MAPHG, DIADIEM)

VALUES (5, N'Nha Trang')

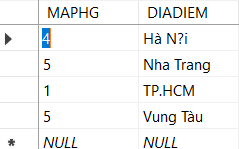
INSERT INTO DIADIEM\_PHG (MAPHG, DIADIEM)

VALUES (5, N'TP.HCM')

ALTER TABLE DIADIEM\_PHG

CHECK CONSTRAINT ALL

END



--Thêm cột STT vào bảng PHANCONG

ALTER TABLE PHANCONG ADD STT INT

BEGIN /\*\* PHANCONG \*\*/

ALTER TABLE PHANCONG

NOCHECK CONSTRAINT ALL

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('009', 1, 1, 32)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('009', 2, 2, 8)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('004', 3, 1, 40)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('003', 1, 2, 20.0)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('003', 2, 1, 20.0)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('008', 10, 1, 35)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('008', 30, 2, 5)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('001', 30, 1, 20)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('001', 20, 1, 15)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('006', 20, 1, 30)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('005', 3, 1, 10)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('005', 10, 2, 10)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('005', 20, 1, 10)

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('007', 30, 2, 30)

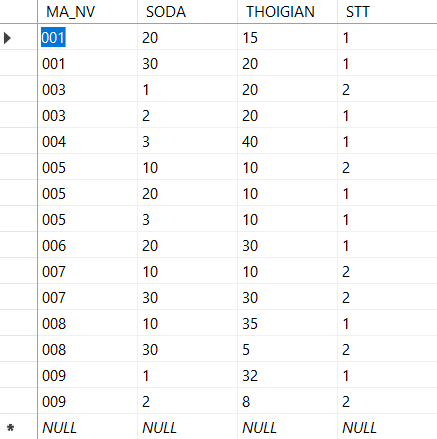
INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('007', 10, 2, 10)

ALTER TABLE PHANCONG

CHECK CONSTRAINT ALL

END



### 2. Hiệu chỉnh kiểu dữ liệu cho cột TENPHG thành nvarchar(20). Thêm vào bảng PhongBan dòng sau:

**2 Nhân sự 10 2003/12/12**

🡪 Dữ liệu đã được thêm vào cột.

-- hiệu chỉnh kiểu dữ liệu cho cột TENPHG thành nvarchar(20)

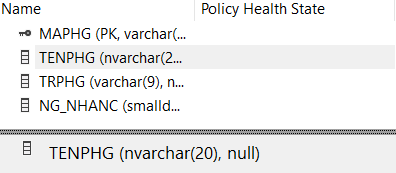
ALTER TABLE PHONGBAN

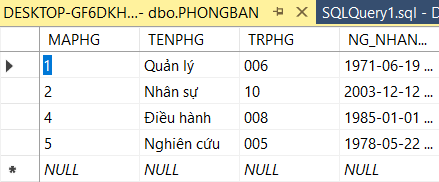
ALTER COLUMN TENPHG NVARCHAR(20);

--thêm dòng mới vào bảng PHONGBAN

INSERT INTO PHONGBAN

VALUES (2, N'Nhân sự', 10, '2003/12/12');



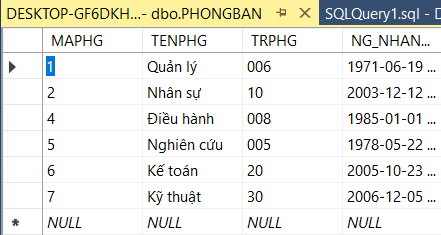


### 3. Thực hiện thêm tiếp các record sau:

INSERT INTO PHONGBAN

VALUES (6, N'Kế toán', 20, '2005/10/23'),

(7, N'Kỹ thuật', 30, '2006/12/05');



### 4. Thêm tùy ý vào bảng DeAn các record có mã đề án (MADA) là 8,9,11,12,13,14,15. Dữ liệu các cột còn lại tùy chọn

INSERT INTO DEAN

VALUES

('08', N'Khảo sát', N'Bến Tre', '1'),

('09', N'Bảo vệ nguồn nước', N'Lào Cai', '2'),

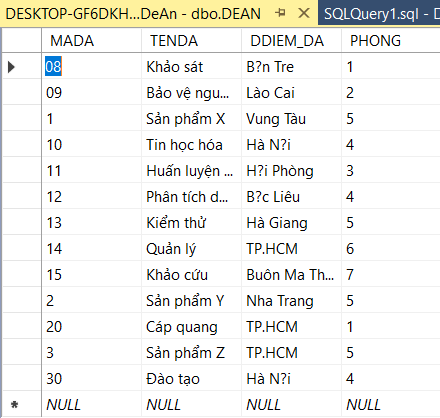
('11', N'Huấn luyện mô hình', N'Hải Phòng', '3'),

('12', N'Phân tích dữ liệu', N'Bạc Liêu', '4'),

('13', N'Kiểm thử', N'Hà Giang', '5'),

('14', N'Quản lý', N'TP.HCM', '6'),

('15', N'Khảo cứu', N'Buôn Ma Thuộc', '7');



### 5. Thêm vào bảng phân công (PHANCONG) 50 dòng cho các nhân viên có mã số từ 10🡪50 tương ứng với các đề án 8,9,11,12,13,14,15 (nhập bằng cửa sổ Edit top 200 rows, không cần dùng lệnh)

- Nhập dữ liệu nhân viên.

INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, PHAI)

VALUES ('Ngô', 'Thị', 'Anh', '010', 'Nữ'),

('Lý', 'Văn', 'Bảo', '011', 'Nam'),

('Lê', 'Văn', 'Phú', '012', 'Nam'),

('Lê', 'Văn', 'Quang', '013', 'Nam'),

('Phạm', 'Thị', 'Út', '014', 'Nữ'),

('Hoàng', 'Văn', 'Sơn', '015', 'Nam'),

('Nguyễn', 'Văn', 'An', '016', 'Nam'),

('Trần', 'Thị', 'Bích', '017', 'Nữ'),

('Lê', 'Văn', 'Cường', '018', 'Nam'),

('Phạm', 'Thị', 'Dung', '019', 'Nữ'),

('Hoàng', 'Văn', 'Dũng', '020', 'Nam'),

('Bùi', 'Thị', 'Loan', '021', 'Nữ'),

('Đặng', 'Văn', 'Giang', '022', 'Nam'),

('Vũ', 'Thị', 'Hương', '023', 'Nữ'),

('Phan', 'Văn', 'Ích', '024', 'Nam'),

('Võ', 'Thị', 'Hồng', '025', 'Nữ'),

('Huỳnh', 'Văn', 'Khang', '026', 'Nam'),

('Ngô', 'Thị', 'Linh', '027', 'Nữ'),

('Lý', 'Văn', 'Minh', '028', 'Nam'),

('Hồ', 'Thị', 'Nga', '029', 'Nữ'),

('Nguyễn', 'Văn', 'Ông', '030', 'Nam'),

('Trần', 'Thị', 'Phương', '031', 'Nữ'),

('Lê', 'Văn', 'Quang', '032', 'Nam'),

('Phạm', 'Thị', 'Út', '033', 'Nữ'),

('Hoàng', 'Văn', 'Sơn', '034', 'Nam'),

('Bùi', 'Thị', 'Thảo', '035', 'Nữ'),

('Đặng', 'Văn', 'Út', '036', 'Nam'),

('Vũ', 'Thị', 'Vân', '037', 'Nữ'),

('Phan', 'Văn', 'Xuân', '038', 'Nam'),

('Võ', 'Thị', 'Yến', '039', 'Nữ'),

('Huỳnh', 'Văn', 'Lữ', '040', 'Nam'),

('Ngô', 'Thị', 'Anh', '041', 'Nữ'),

('Lý', 'Văn', 'Bảo', '042', 'Nam'),

('Hồ', 'Thị', 'Cẩm', '043', 'Nữ'),

('Nguyễn', 'Văn', 'Dũng', '044', 'Nam'),

('Trần', 'Thị', 'Ru', '045', 'Nữ'),

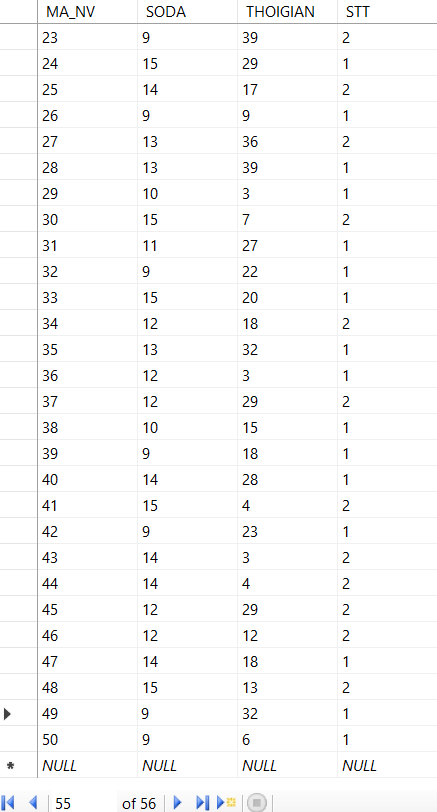
('Lê', 'Văn', 'Phú', '046', 'Nam'),

('Phạm', 'Thị', 'Gấm', '047', 'Nữ'),

('Hoàng', 'Văn', 'Hải', '048', 'Nam'),

('Bùi', 'Thị', 'Vy', '049', 'Nữ'),

('Đặng', 'Văn', 'Đông', '050', 'Nam');

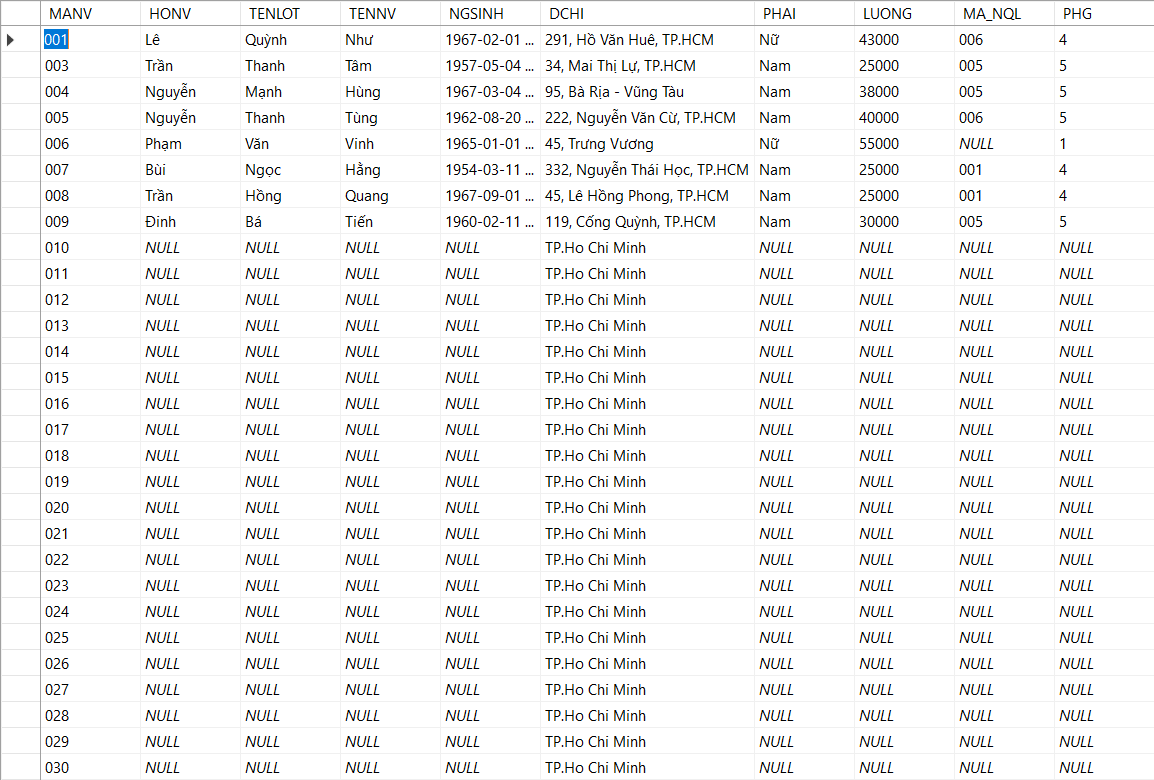


### 6. Thêm dữ liệu vào các cột manv, honv, tenlot, tennv của bảng nhanvien với dữ liệu lấy từ các cột customerID,firstname, middlename, lastname của các nhân viên có mã số từ 102 đến 200 lấy từ bảng Customer của CSDL AdventureWorks.

## Phần 2: câu lệnh UPDATE

### Cập nhật địa chỉ cho các nhân viên có địa chỉ =null là Tp.Hồ Chí Minh trong table Nhanvien

update NHANVIEN Set DCHI = 'TP.Ho Chi Minh' where DCHI is null;



### 2. Cập nhật PHG =2 cho những nhân viên có mã số từ 1 đến 10; PHG=6 cho các nhân viên có mã số từ 11 đến 40, PHG=7 cho các nhân viên có mã số từ 41 đến 80, PHG =5 cho các nhân viên có mã số từ 102🡪200

update NHANVIEN set PHG = case

when MANV between 1 and 10 then 2

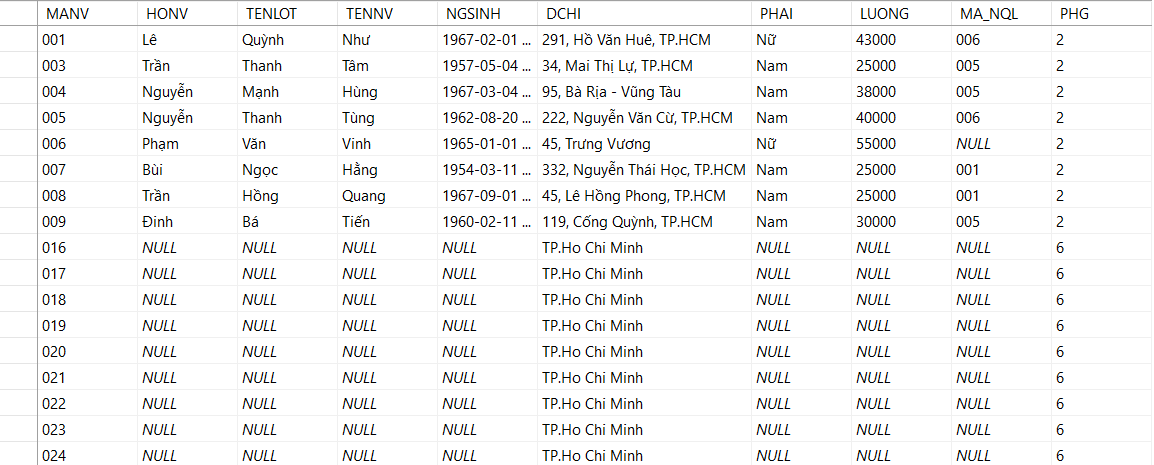
when MANV between 11 and 40 then 6

when MANV between 41 and 80 then 7

when MANV between 102 and 200 then 5

else PHG

end;





### 3. Cập nhật ngày sinh cho những nhân viên có ngày sinh = null (is null) là ngày 01/01/1985 trong table Nhanvien.

update NHANVIEN set NGSINH = '2003-06-06' where NGSINH is null;



### 4. Cập nhật LUONG =100000 cho các nhân viên ở phòng NghienCuu, LUONG=65000 cho các nhân viên ở phòng Kế toán, LUONG=85000 cho các nhân viên ở phòng Kỹ thuật, LUONG = 70000 cho các nhân viên ở phòng nhân sự

UPDATE NHANVIEN

SET LUONG = CASE

WHEN PHG IN (SELECT MAPHG FROM PHONGBAN WHERE TENPHG = N'Nghiên cứu') THEN 100000

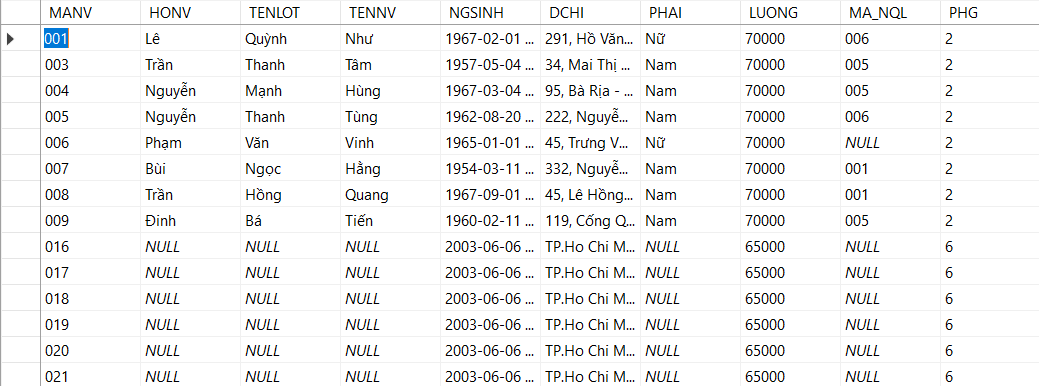
WHEN PHG IN (SELECT MAPHG FROM PHONGBAN WHERE TENPHG = N'Kế toán') THEN 65000

WHEN PHG IN (SELECT MAPHG FROM PHONGBAN WHERE TENPHG = N'Kỹ thuật') THEN 85000

WHEN PHG IN (SELECT MAPHG FROM PHONGBAN WHERE TENPHG = N'Nhân sự') THEN 70000

ELSE LUONG

END;





### 5. Cập nhật mã người quản lý (MA\_NQL) trong bảng nhân viên tương ứng với phòng ban mà nhân viên đó thuộc về. VD: nhân viên có mã nv =8, thuộc phòng ban 2 thì mã người quản lý là 10.

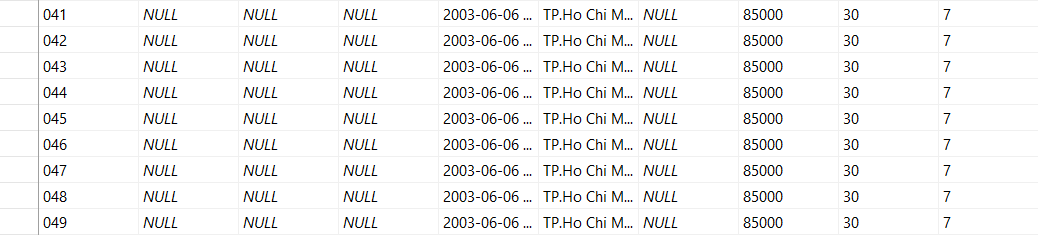
UPDATE NHANVIEN

SET MA\_NQL = PHONGBAN.TRPHG

FROM NHANVIEN

INNER JOIN PHONGBAN ON NHANVIEN.PHG = PHONGBAN.MAPHG





# Phần 3: câu lệnh SELECT

## 3.1 TRUY VẤN CƠ BẢN

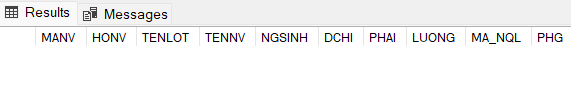
### 3.1.1 Truy vấn đơn giản

##### 1. Tìm các nhân viên làm việc ở phòng số 5

SELECT \*

FROM NHANVIEN

WHERE PHG = 5



- Ở đây không có nhân viên nào ở phòng số 5.

##### Tìm các nhân viên có mức lương trên 25000

SELECT \*

FROM NHANVIEN

WHERE LUONG > 25000

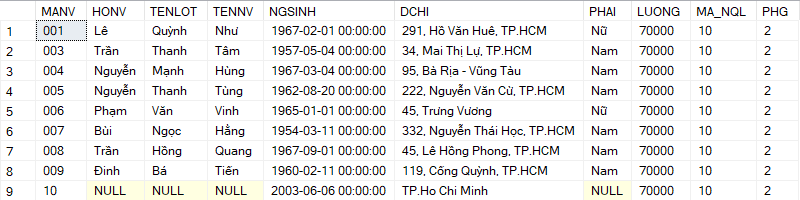


##### Tìm các nhân viên có mức lương trên 65,000 ở phòng 1 hoặc các nhân viên có mức lương trên 70,000 ở phòng 2

SELECT \*

FROM NHANVIEN

WHERE (LUONG >= 65000 AND PHG = 1) OR (LUONG >= 70000 AND PHG = 2)



##### 4. Cho biết họ tên đầy đủ của các nhân viên ở TP.HCM

SELECT HONV + ' ' + TENLOT + ' ' + TENNV AS 'Họ và Tên'

FROM NHANVIEN

WHERE DCHI LIKE N'%TP.HCM%'



##### 5. Cho biết họ tên đầy đủ của các nhân viên có họ bắt đầu bằng ký tự 'N'

SELECT HONV + ' ' + TENLOT + ' ' + TENNV AS 'Họ và Tên'

FROM NHANVIEN

WHERE HONV LIKE N'N%'



##### 6. Cho biết ngày sinh và địa chỉ của nhân viên Lê Minh Tính

SELECT NGSINH,DCHI FROM NHANVIEN WHERE HONV = N'Lê' AND TENLOT = N'Minh' AND TENNV = N'Tính'



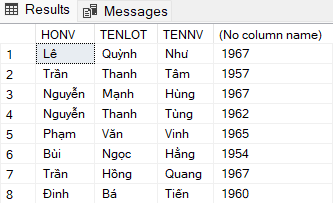
##### 7. Cho biết các nhân viên có năm sinh trong khoảng 1955 đến 1975

SELECT HONV, TENLOT, TENNV FROM NHANVIEN WHERE NGSINH BETWEEN '1955' AND '1975'



##### 8. Cho biết các nhân viên và năm sinh của nhân viên

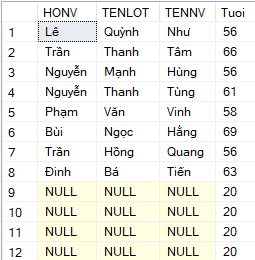
SELECT HONV, TENLOT, TENNV, YEAR(NGSINH) FROM NHANVIEN



##### 9. Cho biết các nhân viên và tuổi của nhân viên

SELECT HONV,TENLOT,TENNV,YEAR('2023/11/20') - YEAR(NGSINH) AS Tuoi

FROM NHANVIEN

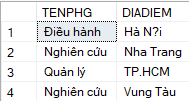


### 3.1.2 Truy vấn có sử dụng phép kết

##### 10.Với mỗi phòng ban, cho biết tên phòng ban và địa điểm phòng

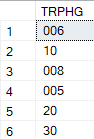
SELECT PHONGBAN.TENPHG, DIADIEM\_PHG.DIADIEM FROM PHONGBAN

INNER JOIN DIADIEM\_PHG ON PHONGBAN.MAPHG = DIADIEM\_PHG.MAPHG



##### 11.Tìm tên những người trưởng phòng của từng phòng ban

SELECT TRPHG FROM PHONGBAN



##### 12.Tìm tên và địa chỉ của tất cả các nhân viên của phòng "Điều hành".

SELECT HONV,TENLOT,TENNV, DCHI FROM PHONGBAN

INNER JOIN NHANVIEN ON NHANVIEN.PHG=PHONGBAN.MAPHG

WHERE TENPHG = N'Điều hành'



##### 13.Với mỗi đề án ở Nha Trang, cho biết tên đề án, tên phòng ban, họ tên và ngày nhận chức của trưởng phòng của phòng ban chủ trì đề án đó.

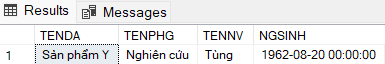
SELECT DEAN.TENDA, PHONGBAN.TENPHG, NHANVIEN.TENNV, NHANVIEN.NGSINH

FROM DEAN

INNER JOIN PHONGBAN ON DEAN.PHONG = PHONGBAN.MAPHG

INNER JOIN NHANVIEN ON PHONGBAN.TRPHG = NHANVIEN.MANV

WHERE DEAN.DDIEM\_DA = N'Nha Trang';



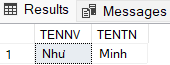
##### 14.Tìm tên những nữ nhân viên và tên người thân của họ

SELECT NHANVIEN.TENNV, THANNHAN.TENTN

FROM NHANVIEN

JOIN THANNHAN ON NHANVIEN.MANV = THANNHAN.MA\_NVIEN

WHERE NHANVIEN.PHAI = N'Nữ'



##### 15.Với mỗi nhân viên, cho biết họ tên nhân viên và họ tên người quản lý trực tiếp của nhân viên đó.

SELECT HONV,TENLOT,TENNV,TRPHG FROM NHANVIEN

INNER JOIN PHONGBAN ON NHANVIEN.PHG=PHONGBAN.MAPHG



##### 16.Với mỗi nhân viên, cho biết họ tên của nhân viên đó, họ tên người trưởng phòng và họ tên người quản lý trực tiếp của nhân viên đó.

SELECT HONV, TENLOT, TENNV, MA\_NQL, TRPHG FROM NHANVIEN

INNER JOIN PHONGBAN ON NHANVIEN.PHG=PHONGBAN.MAPHG



##### 17.Tên những nhân viên phòng số 5 có tham gia vào đề án "Xây dựng nhà máy chế biến thủy sản" và tên người quản lý trực tiếp.

SELECT HONV, TENLOT,TENNV, TRPHG FROM NHANVIEN

INNER JOIN PHONGBAN ON NHANVIEN.PHG = PHONGBAN.MAPHG

INNER JOIN DEAN ON PHONGBAN.MAPHG = DEAN.PHONG

WHERE PHG = 6 AND TENDA = N'Quản lý'



##### 18.Cho biết tên các đề án mà nhân viên Trần Anh Tuấn đã tham gia

SELECT DEAN.TENDA AS 'Tên đề án'

FROM DEAN

INNER JOIN PHANCONG ON DEAN.MADA = PHANCONG.SODA

INNER JOIN NHANVIEN ON PHANCONG.MA\_NV = NHANVIEN.MANV

WHERE NHANVIEN.HONV = N'Trần' AND NHANVIEN.TENLOT = N'Anh' AND NHANVIEN.TENNV = N'Tuấn';

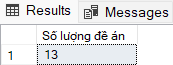


## 3.2 GOM NHÓM

### 19.Cho biết số lượng đề án của công ty

SELECT COUNT(\*) AS 'Số lượng đề án'

FROM DEAN;

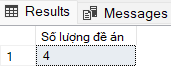


### 20.Cho biết số lượng đề án do phòng 'Nghiên Cứu' chủ trì

SELECT COUNT(\*) AS 'Số lượng đề án'

FROM DEAN

WHERE PHONG IN (SELECT MAPHG FROM PHONGBAN WHERE TENPHG = N'Nghiên cứu')

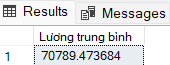


### 21.Cho biết lương trung bình của các nữ nhân viên

SELECT AVG(LUONG) AS 'Lương trung bình'

FROM NHANVIEN

WHERE PHAI = N'Nữ'



### 22.Cho biết số thân nhân của nhân viên ‘Lê Thanh Sang'

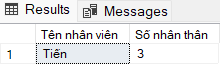
SELECT NHANVIEN.TENNV AS 'Tên nhân viên', COUNT(THANNHAN.MA\_NVIEN) AS 'Số nhân thân'

FROM NHANVIEN

JOIN THANNHAN ON NHANVIEN.MANV = THANNHAN.MA\_NVIEN

WHERE NHANVIEN.HONV = N'Đinh' AND NHANVIEN.TENLOT = N'Bá' AND NHANVIEN.TENNV = N'Tiến'

GROUP BY NHANVIEN.TENNV;



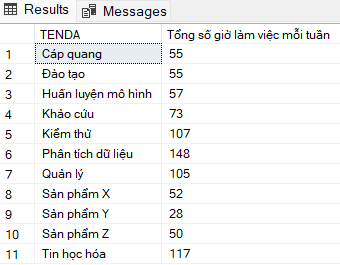
### 23.Với mỗi đề án, liệt kê tên đề án và tổng số giờ làm việc một tuần của tất cả các nhân viên tham dự đề án đó.

SELECT DEAN.TENDA, SUM(PHANCONG.THOIGIAN) AS 'Tổng số giờ làm việc mỗi tuần'

FROM DEAN, PHANCONG

WHERE DEAN.MADA = PHANCONG.SODA

GROUP BY DEAN.TENDA



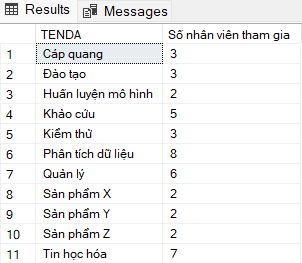
### 24.Với mỗi đề án, cho biết có bao nhiêu nhân viên tham gia đề án đó

SELECT DEAN.TENDA, COUNT(DISTINCT PHANCONG.MA\_NV) AS 'Số nhân viên tham gia'

FROM DEAN, PHANCONG

WHERE DEAN.MADA = PHANCONG.SODA

GROUP BY DEAN.TENDA



### 25.Với mỗi nhân viên, cho biết họ và tên nhân viên và số lượng thân nhân của nhân viên đó.

SELECT NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV, COUNT(THANNHAN.MA\_NVIEN) AS 'Số lượng thân nhân'

FROM NHANVIEN LEFT JOIN THANNHAN

ON NHANVIEN.MANV = THANNHAN.MA\_NVIEN

GROUP BY NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV



### 26.Với mỗi nhân viên, cho biết họ tên của nhân viên và số lượng đề án mà nhân viên đó đã tham gia.

SELECT NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV, COUNT(DISTINCT PHANCONG.SODA) AS 'Số lượng đề án tham gia'

FROM NHANVIEN LEFT JOIN PHANCONG

ON NHANVIEN.MANV = PHANCONG.MA\_NV

GROUP BY NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV



### 27.Với mỗi nhân viên, cho biết số lượng nhân viên mà nhân viên đó quản lý trực tiếp.

SELECT N1.HONV, N1.TENLOT, N1.TENNV, COUNT(N2.MANV) AS 'Số lượng nhân viên quản lý'

FROM NHANVIEN N1 LEFT JOIN NHANVIEN N2

ON N1.MANV = N2.MA\_NQL

GROUP BY N1.HONV, N1.TENLOT, N1.TENNV



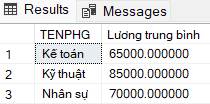
### 28.Với mỗi phòng ban, liệt kê tên phòng ban và lương trung bình của những nhân viên làm việc cho phòng ban đó.

SELECT PHONGBAN.TENPHG, AVG(NHANVIEN.LUONG) AS 'Lương trung bình'

FROM PHONGBAN, NHANVIEN

WHERE PHONGBAN.MAPHG = NHANVIEN.PHG

GROUP BY PHONGBAN.TENPHG



### 29.Với các phòng ban có mức lương trung bình trên 30,000, liệt kê tên phòng ban và số lượng nhân viên của phòng ban đó.

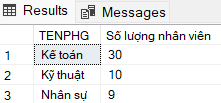
SELECT PHONGBAN.TENPHG, COUNT(NHANVIEN.MANV) AS 'Số lượng nhân viên'

FROM PHONGBAN JOIN NHANVIEN

ON PHONGBAN.MAPHG = NHANVIEN.PHG

GROUP BY PHONGBAN.TENPHG

HAVING AVG(NHANVIEN.LUONG) > 30000



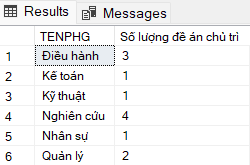
### 30.Với mỗi phòng ban, cho biết tên phòng ban và số lượng đề án mà phòng ban đó chủ trì

SELECT PHONGBAN.TENPHG, COUNT(DEAN.MADA) AS 'Số lượng đề án chủ trì'

FROM PHONGBAN LEFT JOIN DEAN

ON PHONGBAN.MAPHG = DEAN.PHONG

GROUP BY PHONGBAN.TENPHG



### 31.Với mỗi phòng ban, cho biết tên phòng ban, họ tên người trưởng phòng và số lượng đề án mà phòng ban đó chủ trì

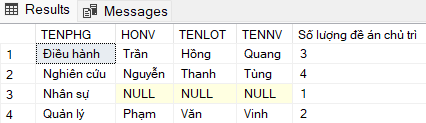
SELECT PHONGBAN.TENPHG, NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV, COUNT(DEAN.MADA) AS 'Số lượng đề án chủ trì'

FROM PHONGBAN

JOIN NHANVIEN ON PHONGBAN.TRPHG = NHANVIEN.MANV

LEFT JOIN DEAN ON PHONGBAN.MAPHG = DEAN.PHONG

GROUP BY PHONGBAN.TENPHG, NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV



### 32.Với mỗi phòng ban có mức lương trung bình lớn hơn 50,000, cho biết tên phòng ban và số lượng đề án mà phòng ban đó chủ trì.

SELECT PHONGBAN.TENPHG, COUNT(DEAN.MADA) AS 'Số lượng đề án chủ trì'

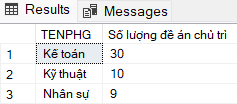
FROM PHONGBAN

JOIN NHANVIEN ON PHONGBAN.MAPHG = NHANVIEN.PHG

LEFT JOIN DEAN ON PHONGBAN.MAPHG = DEAN.PHONG

GROUP BY PHONGBAN.TENPHG

HAVING AVG(NHANVIEN.LUONG) > 50000



### 33.Cho biết số đề án diễn ra tại từng địa điểm

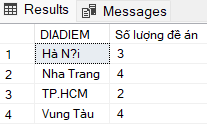
SELECT DIADIEM\_PHG.DIADIEM, COUNT(DEAN.MADA) AS 'Số lượng đề án'

FROM DIADIEM\_PHG

JOIN PHONGBAN ON DIADIEM\_PHG.MAPHG = PHONGBAN.MAPHG

JOIN DEAN ON PHONGBAN.MAPHG = DEAN.PHONG

GROUP BY DIADIEM\_PHG.DIADIEM



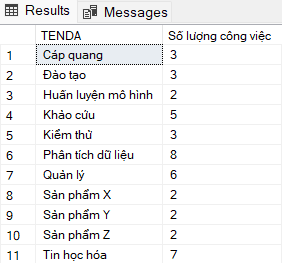
### 34.Với mỗi đề án, cho biết tên đề án và số lượng công việc của đề án này

SELECT DEAN.TENDA, COUNT(PHANCONG.MA\_NV) AS 'Số lượng công việc'

FROM DEAN

JOIN PHANCONG ON DEAN.MADA = PHANCONG.SODA

GROUP BY DEAN.TENDA



## 3.3 TRUY VẤN LỒNG + GOM NHÓM

### 36.Cho biết danh sách các đề án (MADA) có: nhân công với họ (HONV) là 'Lê' hoặc có người trưởng phòng chủ trì đề án với họ (HONV) là 'Lê'.

SELECT DEAN.MADA AS 'Mã đề án'

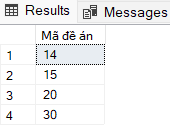
FROM DEAN

JOIN PHANCONG ON DEAN.MADA = PHANCONG.SODA

JOIN NHANVIEN ON PHANCONG.MA\_NV = NHANVIEN.MANV OR DEAN.PHONG = NHANVIEN.PHG

WHERE NHANVIEN.HONV = N'Lê'

GROUP BY DEAN.MADA



### 37.Danh sách những nhân viên (HONV, TENLOT, TENNV) có trên 2 thân nhân.

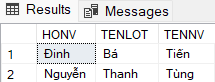
SELECT NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV

FROM NHANVIEN

JOIN THANNHAN ON NHANVIEN.MANV = THANNHAN.MA\_NVIEN

GROUP BY NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV

HAVING COUNT(THANNHAN.MA\_NVIEN) > 2



### 38.Danh sách những nhân viên (HONV, TENLOT, TENNV) không có thân nhân nào (dùng NOT IN, LEFT JOIN, NOT EXISTS)

--1. Sử dụng NOT IN:

SELECT NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV

FROM NHANVIEN

WHERE NHANVIEN.MANV NOT IN (SELECT THANNHAN.MA\_NVIEN FROM THANNHAN)

--2. Sử dụng LEFT JOIN:

SELECT NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV

FROM NHANVIEN

LEFT JOIN THANNHAN ON NHANVIEN.MANV = THANNHAN.MA\_NVIEN

WHERE THANNHAN.MA\_NVIEN IS NULL

--3. Sử dụng NOT EXISTS:

SELECT NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV

FROM NHANVIEN

WHERE NOT EXISTS (SELECT 1 FROM THANNHAN WHERE NHANVIEN.MANV = THANNHAN.MA\_NVIEN)



### 39.Danh sách những trưởng phòng (HONV, TENLOT, TENNV) có tối thiểu một thân nhân.

SELECT NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV

FROM NHANVIEN

JOIN PHONGBAN ON NHANVIEN.MANV = PHONGBAN.TRPHG

WHERE EXISTS (SELECT 1 FROM THANNHAN WHERE NHANVIEN.MANV = THANNHAN.MA\_NVIEN)



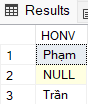
### 40.Tìm họ (HONV) của những trưởng phòng chưa có gia đình.

SELECT NHANVIEN.HONV

FROM NHANVIEN

JOIN PHONGBAN ON NHANVIEN.MANV = PHONGBAN.TRPHG

WHERE NOT EXISTS (SELECT 1 FROM THANNHAN WHERE NHANVIEN.MANV = THANNHAN.MA\_NVIEN)



### 41.Cho biết họ tên nhân viên (HONV, TENLOT, TENNV) có mức lương trên mức lương trung bình của phòng "Nhân sự"

SELECT N1.HONV, N1.TENLOT, N1.TENNV

FROM NHANVIEN N1

WHERE N1.LUONG > (

SELECT AVG(N2.LUONG)

FROM NHANVIEN N2

JOIN PHONGBAN ON N2.PHG = PHONGBAN.MAPHG

WHERE PHONGBAN.TENPHG = N'Nhân sự'

)



### 42.Cho biết tên phòng ban và họ tên trưởng phòng của phòng ban có đông nhân viên nhất.

SELECT TOP 1 PHONGBAN.TENPHG, NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV, COUNT(\*) AS 'Số lượng nhân viên'

FROM PHONGBAN

JOIN NHANVIEN ON PHONGBAN.MAPHG = NHANVIEN.PHG

GROUP BY PHONGBAN.TENPHG, NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV

ORDER BY COUNT(\*) DESC



### 43.Cho biết danh sách các mã đề án mà nhân viên có mã là 60 chưa làm.

SELECT DEAN.MADA

FROM DEAN

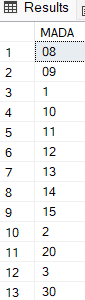
WHERE DEAN.MADA NOT IN (

SELECT PHANCONG.SODA

FROM PHANCONG

WHERE PHANCONG.MA\_NV = '60'

)



### 44.Danh sách nhân viên gồm MaNv, HovaTen (được nối từ các cột HONV, TENLOT, TENNV) và địa chỉ (DCHI) của những nhân viên làm việc cho một đề án ở 'TP HCM' nhưng phòng ban mà họ trực thuộc lại không tọa lạc ở thành phố 'TP HCM' .

SELECT NHANVIEN.MANV 'Mã nhân viên', CONCAT(NHANVIEN.HONV, ' ', NHANVIEN.TENLOT, ' ', NHANVIEN.TENNV) AS 'Họ và Tên', NHANVIEN.DCHI AS 'Địa chỉ'

FROM NHANVIEN

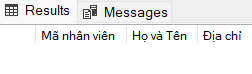
JOIN PHANCONG ON NHANVIEN.MANV = PHANCONG.MA\_NV

JOIN DEAN ON PHANCONG.SODA = DEAN.MADA

JOIN PHONGBAN ON NHANVIEN.PHG = PHONGBAN.MAPHG

JOIN DIADIEM\_PHG ON PHONGBAN.MAPHG = DIADIEM\_PHG.MAPHG

WHERE DEAN.DDIEM\_DA = N'TP.HCM' AND DIADIEM\_PHG.DIADIEM <> N'TP.HCM'



### 45. Tổng quát câu 16, tìm họ tên và địa chỉ của các nhân viên làm việc cho một đề án ở một thành phố nhưng phòng ban mà họ trực thuộc lại không toạ lạc ở thành phố đó.

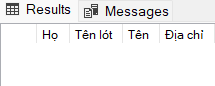
SELECT NHANVIEN.HONV 'Họ', NHANVIEN.TENLOT 'Tên lót', NHANVIEN.TENNV 'Tên', NHANVIEN.DCHI 'Địa chỉ'

FROM NHANVIEN

INNER JOIN DEAN ON NHANVIEN.PHG = DEAN.PHONG

INNER JOIN DIADIEM\_PHG ON DEAN.PHONG = DIADIEM\_PHG.MAPHG

WHERE DEAN.DDIEM\_DA != DIADIEM\_PHG.DIADIEM



## 3.5 COMPUTE/COMPUTE BY

### 46.Danh sách nhân viên gồm MaNv, HovaTen (được nối từ các cột HONV, TENLOT, TENNV), Tuổi và tuổi trung bình của các nhân viên trong công ty (dùng COMPUTE)

SELECT

MANV 'Mã nhân viên',

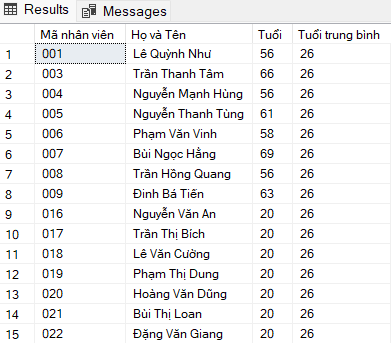
CONCAT(HONV, ' ', TENLOT, ' ', TENNV) AS 'Họ và Tên',

YEAR(CURRENT\_TIMESTAMP) - YEAR(NGSINH) AS 'Tuổi',

(SELECT AVG(YEAR(CURRENT\_TIMESTAMP) - YEAR(NGSINH)) FROM NHANVIEN) AS 'Tuổi trung bình'

FROM NHANVIEN

GROUP BY MANV, HONV, TENLOT, TENNV, NGSINH;



### 47.Danh sách các Phòng ban, Tên đề án và tổng lương trung bình phải trả cho mỗi đề án (COMPUTE BY). Lương trung bình được tính từ trung bình của tổng lương\_đề án các nhân viên, trong đó lương\_đề án của mỗi nhân viên = LUONGx với số giờ làm việc cho mỗi đề án của nhân viên đó.

SELECT PHG.TENPHG 'Tên phòng', DA.TENDA 'Tên đề án', AVG(NV.LUONG \* PHANCONG.THOIGIAN) AS 'Lương trung bình'

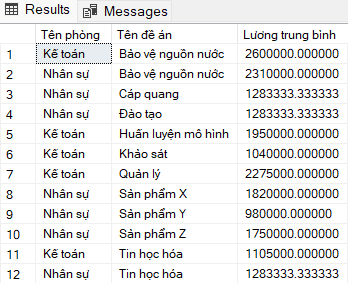
FROM PHONGBAN PHG

JOIN NHANVIEN NV ON PHG.MAPHG = NV.PHG

JOIN PHANCONG ON NV.MANV = PHANCONG.MA\_NV

JOIN DEAN DA ON PHANCONG.SODA = DA.MADA

GROUP BY PHG.TENPHG, DA.TENDA;



## 3.4 PHÉP CHIA

### 48.Danh sách những nhân viên MaNV, Phái, HoTen được nối từ HONV, TENLOT, TENNV làm việc trong mọi đề án của công ty

SELECT NHANVIEN.MANV 'Mã nhân viên', NHANVIEN.PHAI 'Phái', NHANVIEN.HONV + ' ' + NHANVIEN.TENLOT + ' ' + NHANVIEN.TENNV AS 'Họ và Tên'

FROM NHANVIEN

EXCEPT

SELECT NHANVIEN.MANV, NHANVIEN.PHAI, NHANVIEN.HONV + ' ' + NHANVIEN.TENLOT + ' ' + NHANVIEN.TENNV AS HoTen

FROM NHANVIEN

JOIN PHANCONG ON NHANVIEN.MANV = PHANCONG.MA\_NV

GROUP BY NHANVIEN.MANV, NHANVIEN.PHAI, NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV

HAVING COUNT(DISTINCT PHANCONG.SODA) < (SELECT COUNT(\*) FROM DEAN)



### 49.Danh sách những nhân viên MaNV, Phái, HoTen được nối từ HONV, TENLOT, TENNV được phân công tất cả đề án do phòng số 5 chủ trì.

SELECT NHANVIEN.MANV 'Mã nhân viên', NHANVIEN.PHAI 'Phái', NHANVIEN.HONV + ' ' + NHANVIEN.TENLOT + ' ' + NHANVIEN.TENNV AS 'Họ và Tên'

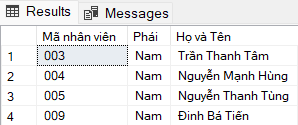
FROM NHANVIEN

JOIN PHANCONG ON NHANVIEN.MANV = PHANCONG.MA\_NV

WHERE PHANCONG.SODA IN (SELECT MADA FROM DEAN WHERE PHONG = 5)

GROUP BY NHANVIEN.MANV, NHANVIEN.PHAI, NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV

HAVING COUNT(DISTINCT PHANCONG.SODA) < (SELECT COUNT(\*) FROM DEAN WHERE PHONG = 5)



### 50.Tìm những nhân viên MaNV, Phái, HoTen (HONV, TENLOT, TENNV) được phân công tất cả đề án mà nhân viên Lê Minh Tính làm việc

SELECT N1.MANV 'Mã nhân viên', N1.PHAI 'Phái', CONCAT(N1.HONV, ' ', N1.TENLOT, ' ', N1.TENNV) AS 'Họ và Tên'

FROM NHANVIEN N1

WHERE (

SELECT COUNT(\*)

FROM PHANCONG PC1

WHERE PC1.MA\_NV = N1.MANV

) = (

SELECT COUNT(\*)

FROM (

SELECT DISTINCT PC2.SODA

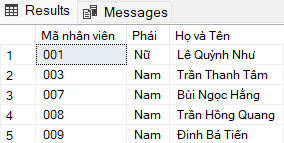
FROM NHANVIEN N2

JOIN PHANCONG PC2 ON N2.MANV = PC2.MA\_NV

WHERE CONCAT(N2.HONV, ' ', N2.TENLOT, ' ', N2.TENNV) = N'Lê Quỳnh Như'

) AS Projects

);



### 51.Cho biết danh sách nhân viên tham gia vào tất cả các đề án ở TP HCM

SELECT N1.MANV 'Mã nhân viên', CONCAT(N1.HONV, ' ', N1.TENLOT, ' ', N1.TENNV) AS 'Họ và Tên'

FROM NHANVIEN N1

WHERE NOT EXISTS (

SELECT D1.MADA

FROM DEAN D1

WHERE D1.DDIEM\_DA = N'Nha Trang'

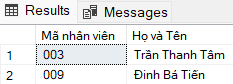
EXCEPT

SELECT PC1.SODA

FROM PHANCONG PC1

WHERE PC1.MA\_NV = N1.MANV

);



### 52.Cho biết phòng ban chủ trì tất cả các đề án ở HaNoi

SELECT PB.MAPHG, PB.TENPHG

FROM PHONGBAN PB

WHERE NOT EXISTS (

SELECT DA.MADA

FROM DEAN DA

INNER JOIN DIADIEM\_PHG DD ON DA.PHONG = DD.MAPHG

WHERE DD.DIADIEM = N'Hà Nội' AND NOT EXISTS (

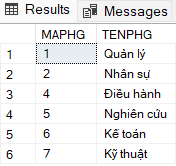
SELECT DA1.MADA

FROM DEAN DA1

WHERE DA1.PHONG = PB.MAPHG AND DA1.MADA = DA.MADA

)

)



## 3.6. UNION/INTERSECT

### 53.Cho biết những phòng ban có nhân viên tham gia cả 2 dự án ở HaNoi và Tp.HCM.

SELECT PB.MAPHG, PB.TENPHG

FROM PHONGBAN PB

WHERE EXISTS (

SELECT NV.MANV

FROM NHANVIEN NV

INNER JOIN PHANCONG PC ON NV.MANV = PC.MA\_NV

INNER JOIN DEAN DA ON PC.SODA = DA.MADA

INNER JOIN DIADIEM\_PHG DD ON DA.PHONG = DD.MAPHG

WHERE DD.DIADIEM = 'Hà Nội' AND NV.PHG = PB.MAPHG

)

INTERSECT

SELECT PB.MAPHG, PB.TENPHG

FROM PHONGBAN PB

WHERE EXISTS (

SELECT NV.MANV

FROM NHANVIEN NV

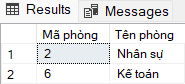
INNER JOIN PHANCONG PC ON NV.MANV = PC.MA\_NV

INNER JOIN DEAN DA ON PC.SODA = DA.MADA

INNER JOIN DIADIEM\_PHG DD ON DA.PHONG = DD.MAPHG

WHERE DD.DIADIEM = 'TP.HCM' AND NV.PHG = PB.MAPHG

)



### 54. Cho biết những phòng ban có nhân viên tham gia dự án ở Tp.HCM hoặc ở HaNoi

SELECT PB.MAPHG, PB.TENPHG

FROM PHONGBAN PB

WHERE EXISTS (

SELECT NV.MANV

FROM NHANVIEN NV

INNER JOIN PHANCONG PC ON NV.MANV = PC.MA\_NV

INNER JOIN DEAN DA ON PC.SODA = DA.MADA

INNER JOIN DIADIEM\_PHG DD ON DA.PHONG = DD.MAPHG

WHERE DD.DIADIEM = N'TP.HCM' AND NV.PHG = PB.MAPHG

)

UNION

SELECT PB.MAPHG, PB.TENPHG

FROM PHONGBAN PB

WHERE EXISTS (

SELECT NV.MANV

FROM NHANVIEN NV

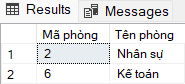
INNER JOIN PHANCONG PC ON NV.MANV = PC.MA\_NV

INNER JOIN DEAN DA ON PC.SODA = DA.MADA

INNER JOIN DIADIEM\_PHG DD ON DA.PHONG = DD.MAPHG

WHERE DD.DIADIEM = N'HaNoi' AND NV.PHG = PB.MAPHG

)



# Module 4 & 5

## PHẦN 1: BATCH

### 1. Viết đoạn batch để in ra tổng số dự án mà nhân viên có MANV = ‘333445555’ tham gia. Nếu tổng số dự án <3 thì in ra “Tham gia ít” ngược lại in ra “ Tham gia nhiều”. Lưu ý, MaNV phải được sử dụng như biến, không gán giá trị trực tiếp vào điều kiện của câu truy vấn.

DECLARE @MaNV varchar(9);

SET @MaNV = '333445555';

DECLARE @ProjectCount int;

SELECT @ProjectCount = COUNT(\*)

FROM PHANCONG

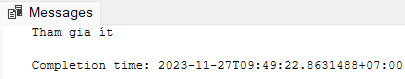
WHERE MA\_NV = @MaNV;

IF @ProjectCount < 3

PRINT 'Tham gia ít'

ELSE

PRINT 'Tham gia nhiều';



### 2. Viết vòng lặp while để xuất ra bảng cửu chương 2

DECLARE @i int;

SET @i = 1;

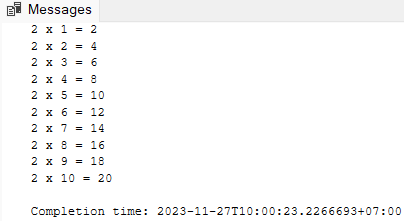
WHILE @i <= 10

BEGIN

PRINT '2 x ' + CAST(@i AS varchar) + ' = ' + CAST((2\*@i) AS varchar);

SET @i = @i + 1;

END



### 3. Thêm một cột Email(varchar(30)) vào bảng NHANVIEN. Viết vòng lặp WHILE để chèn dữ liệu cho cột Email cho các nhân viên theo mẫu sau :

User\_MaNV@mail.com.

Ví dụ, nhân viên có mã số 101 được thêm vào dữ liệu: User\_101@mail.com

-- Thêm cột Email vào bảng NHANVIEN

ALTER TABLE NHANVIEN

ADD EMAIL varchar(30);

-- Khai báo biến

DECLARE @MaNV varchar(9);

DECLARE @Email varchar(30);

SELECT @MaNV = MIN(MANV) FROM NHANVIEN;

-- Chạy vòng lặp hết tất cả các hàng trong bảng NHANVIEN

WHILE @MaNV IS NOT NULL

BEGIN

-- Tạo địa chỉ email

SET @Email = 'User\_' + @MaNV + '@mail.com';

-- Cập nhật cột Email

UPDATE NHANVIEN

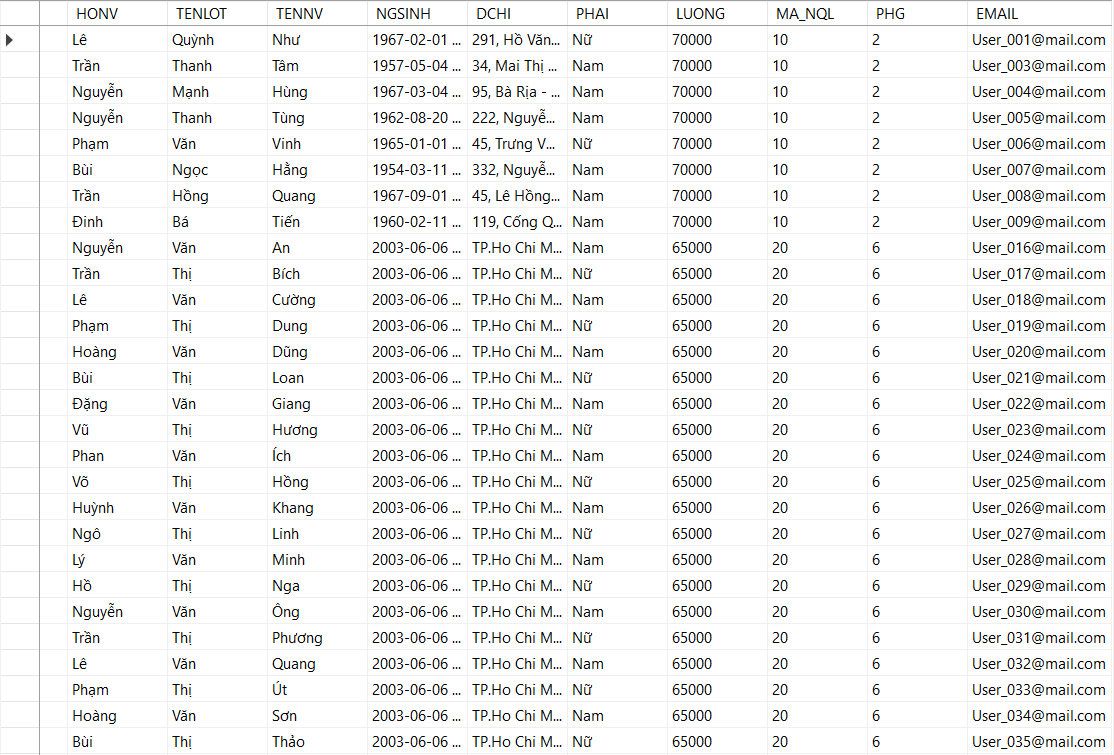
SET EMAIL = @Email

WHERE MANV = @MaNV;

-- Get MANV tiếp theo

SELECT @MaNV = M IN(MANV) FROM NHANVIEN WHERE MANV > @MaNV;

END



### 4. Viết đoạn batch sử dụng SELECT …CASE để trả về thông tin của nhân viên cùng với độ tuổi của nhân viên. Trong đó, độ tuổi tính như sau :nếu tuổi của nhân viên từ 18🡪30 là độ tuổi thanh niên, từ 31 đến 45 là trung niên, từ 46 đến 60 là cao niên.

SELECT

MANV,

HONV,

TENLOT,

TENNV,

NGSINH,

CASE

WHEN DATEDIFF(YEAR, NGSINH, GETDATE()) BETWEEN 18 AND 30 THEN 'Thanh niên'

WHEN DATEDIFF(YEAR, NGSINH, GETDATE()) BETWEEN 31 AND 45 THEN 'Trung niên'

WHEN DATEDIFF(YEAR, NGSINH, GETDATE()) BETWEEN 46 AND 60 THEN 'Cao niên'

ELSE 'Khác'

END AS 'Độ tuổi'

FROM NHANVIEN;



## Phần 2 : VIEWS

### 1. Viết truy vấn liệt kê tên các nhân viên cùng với các dự án mà nhân viên đó tham gia. Run và kiểm tra kết quả.

SELECT NHANVIEN.TENNV 'Tên nhân viên', DEAN.TENDA 'Tên đề án'

FROM NHANVIEN

JOIN PHANCONG ON NHANVIEN.MANV = PHANCONG.MA\_NV

JOIN DEAN ON PHANCONG.SODA = DEAN.MADA;



### 2. Tạo một View có tên NV\_DA với nội dung truy vấn là câu 1. Mở cửa sổ Object Explorer, xem view vừa tạo lưu ở đâu? Xem kết quả dữ liệu từ View vừa tạo (SELECT \* FROM NV\_DA) và so sánh với kết quả ở câu 1; có khác nhau không? Tại sao?

CREATE VIEW NV\_DA AS

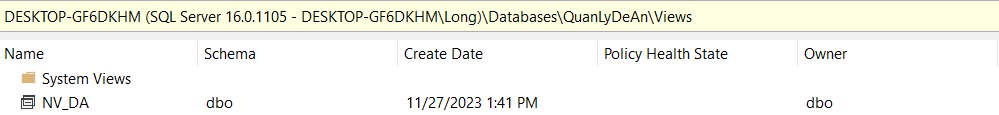
SELECT NHANVIEN.TENNV 'Tên nhân viên', DEAN.TENDA 'Tên đề án'

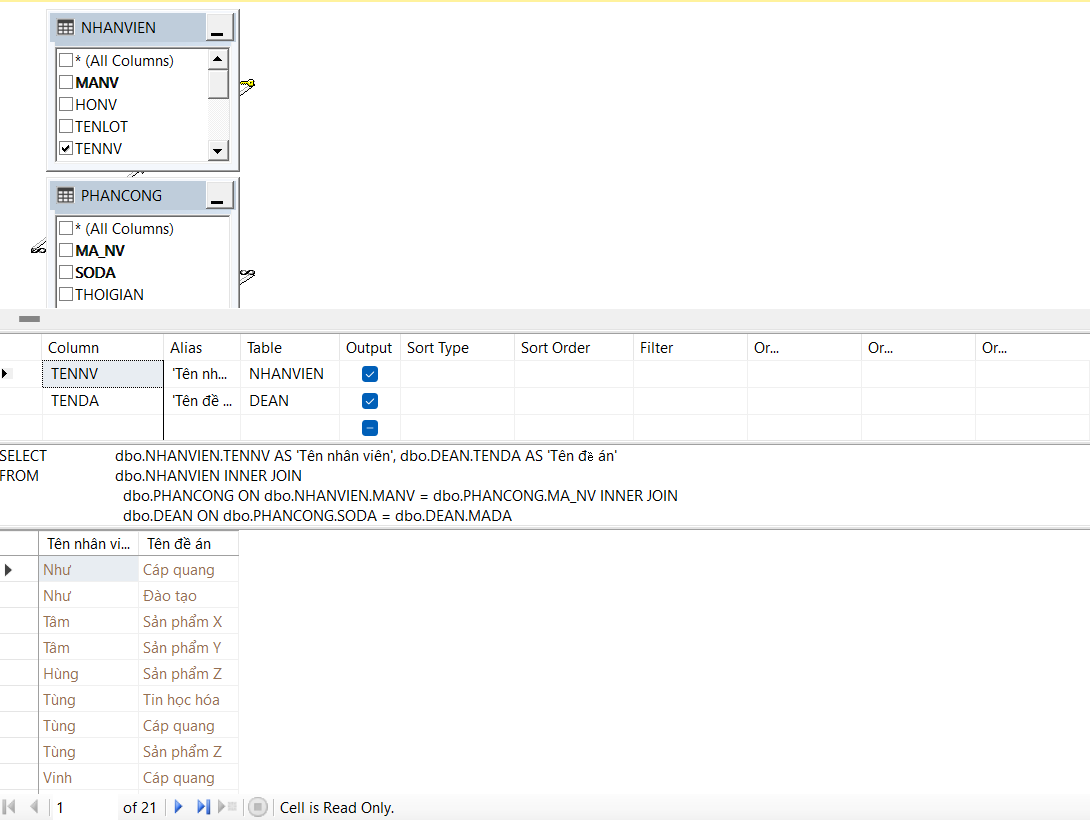
FROM NHANVIEN

JOIN PHANCONG ON NHANVIEN.MANV = PHANCONG.MA\_NV

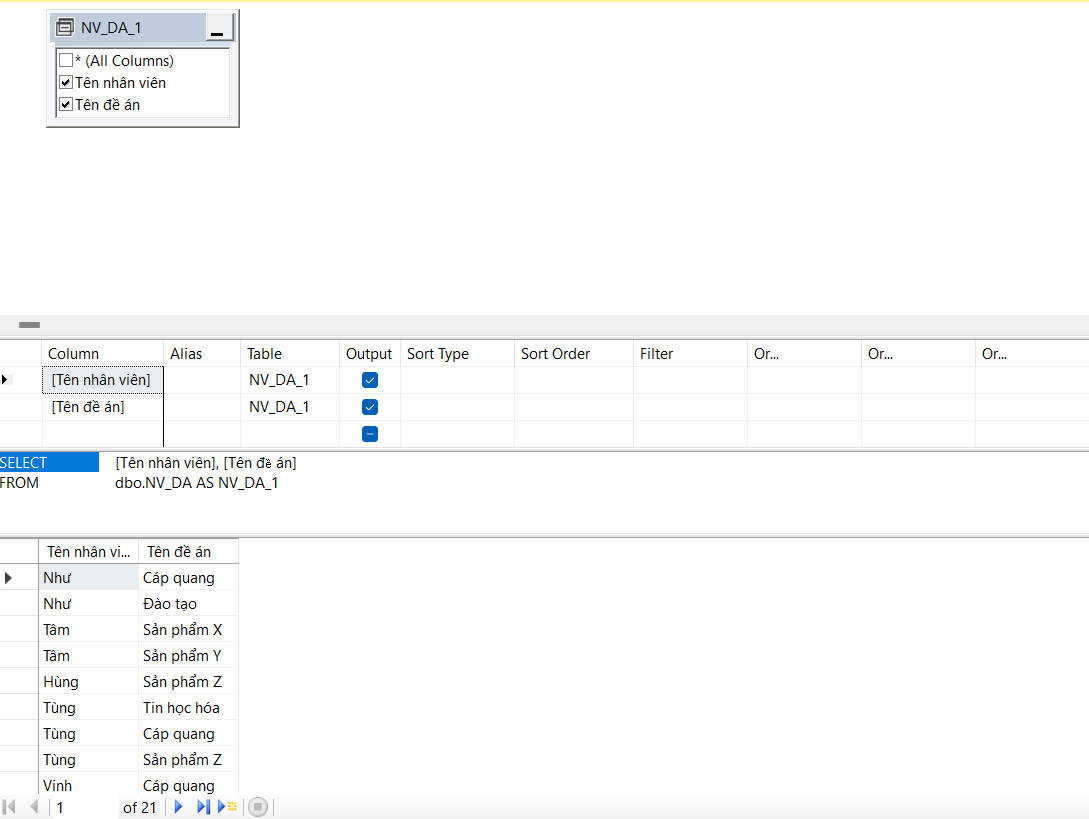
JOIN DEAN ON PHANCONG.SODA = DEAN.MADA;

View đã tạo được lưu ở thư mục Views.





SELECT \* FROM NV\_DA



Kết quả từ SELECT \* FROM NV\_DA và truy vấn đầu tiên sẽ giống nhau. Là bởi vì một view trong SQL là một bảng ảo dựa trên tập kết quả của một câu lệnh SQL, và nó sẽ phản ánh dữ liệu của câu lệnh SQL đã thực thi.

### 3. Thêm một dòng tùy ý vào bảng PHANCONG, sau đó thực hiện lệnh SELECT \* FROM NV\_DA, kết quả có thay đổi so với câu 2 không? Tại sao?

INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, PHAI)

VALUES (N'Trần', N'Văn', N'Huy', '051', 'Nam')

update NHANVIEN Set DCHI = 'TP.Ho Chi Minh' where DCHI is null;

update NHANVIEN set PHG = case

when MANV between 1 and 10 then 2

when MANV between 11 and 40 then 6

when MANV between 41 and 80 then 7

when MANV between 102 and 200 then 5

else PHG

end;

update NHANVIEN set NGSINH = '2003-06-06' where NGSINH is null;

UPDATE NHANVIEN

SET LUONG = CASE

WHEN PHG IN (SELECT MAPHG FROM PHONGBAN WHERE TENPHG = N'Nghiên cứu') THEN 100000

WHEN PHG IN (SELECT MAPHG FROM PHONGBAN WHERE TENPHG = N'Kế toán') THEN 65000

WHEN PHG IN (SELECT MAPHG FROM PHONGBAN WHERE TENPHG = N'Kỹ thuật') THEN 85000

WHEN PHG IN (SELECT MAPHG FROM PHONGBAN WHERE TENPHG = N'Nhân sự') THEN 70000

ELSE LUONG

END;

UPDATE NHANVIEN

SET MA\_NQL = PHONGBAN.TRPHG

FROM NHANVIEN

INNER JOIN PHONGBAN ON NHANVIEN.PHG = PHONGBAN.MAPHG

BEGIN /\*\* PHANCONG \*\*/

ALTER TABLE PHANCONG

NOCHECK CONSTRAINT ALL

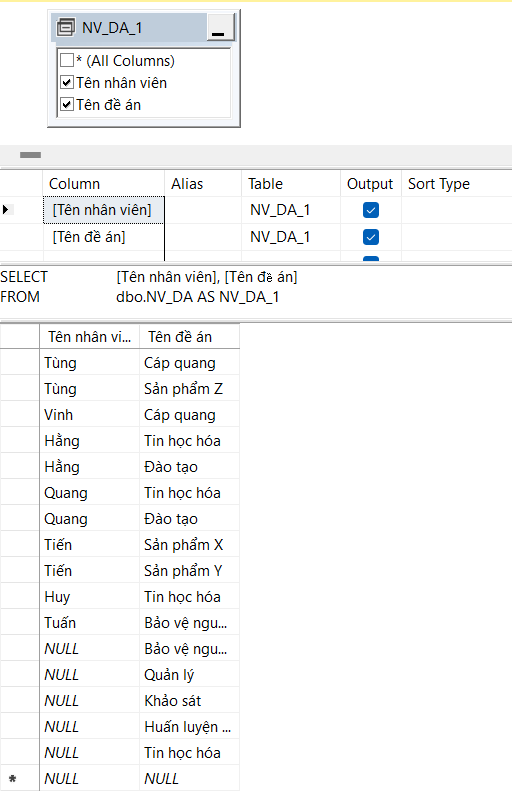
INSERT INTO PHANCONG (MA\_NV, SODA, THOIGIAN, STT)

VALUES ('051', 10, 20, 2)

ALTER TABLE PHANCONG

CHECK CONSTRAINT ALL

END



Kết quả có thay đổi so với câu 2. Do view NV\_DA được tạo từ truy vấn liên kết các bảng NHANVIEN, PHANCONG, và DEAN.

### 4. Thực hiện lệnh:

UPDATE NV\_DA

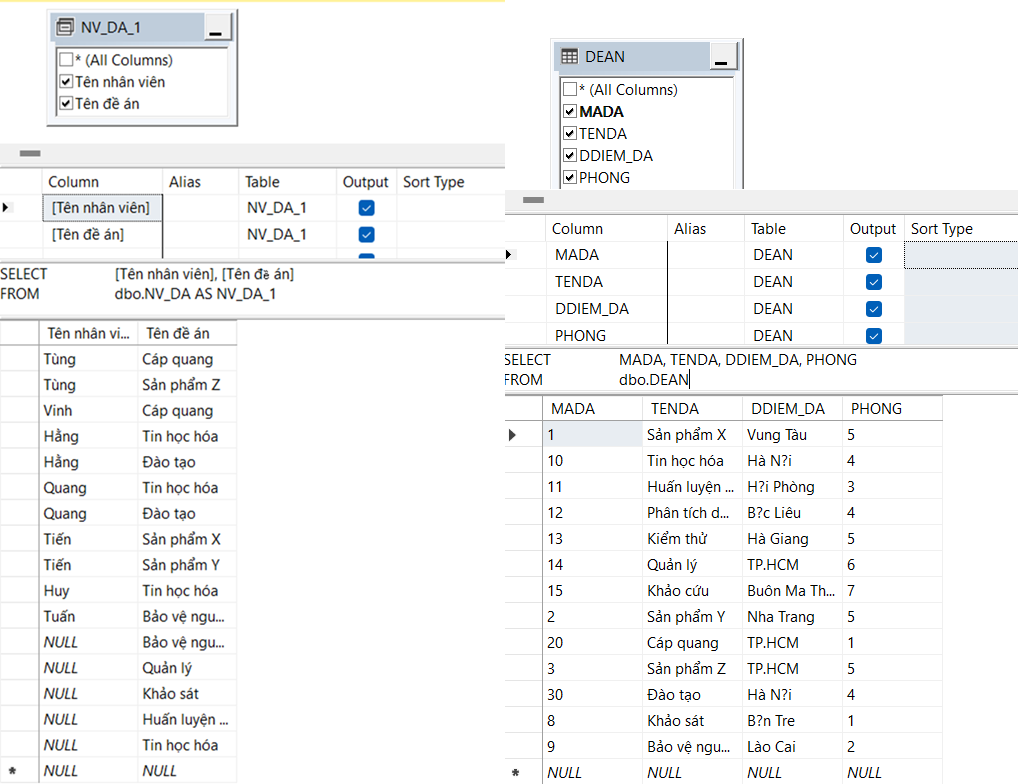
SET TENDA = N'Quản lý các dự án CNTT thông tin'

WHERE TENDA = N'Resort nghỉ dưỡng'

Câu lệnh có thực hiện được không? Tại sao?

* Câu lệnh UPDATE NV\_DA không thể thực hiện được. Lý do là NV\_DA là một view, không phải là một bảng. Trong SQL, một view là một bảng ảo dựa trên kết quả của một câu lệnh SQL. Nó không tồn tại như một bảng thực sự trong cơ sở dữ liệu, vì vậy bạn không thể cập nhật dữ liệu trực tiếp trong view.
* Nếu muốn cập nhật dữ liệu phải cần cập nhật các bảng gốc mà view đang tham chiếu đến.

Thực hiện câu lệnh: SELECT \* FROM NV\_DA và SELECT \* FROM DEAN để xem sự thay đổi dữ liệu.



Ở câu lệnh SELECT \* FROM NV\_DA, câu lệnh sẽ thực hiện truy vấn đã được sử dụng khi tạo nên view. Còn câu lệnh SELECT \* FROM DEAN thì chỉ truy vấn bảng DEAN đã được join trước đó.

### 5. Viết truy vấn hiển thị HoTen của nhân viên, tên dự án nhân viên đó tham gia cùng với tổng tiền lương của nhân viên theo dự án (tiền lương = số giờ \* LUONG). Sắp xếp tiền lương tăng dần. Sau đó, thực hiện các yêu cầu sau:

SELECT CONCAT(NHANVIEN.HONV, ' ', NHANVIEN.TENLOT, ' ', NHANVIEN.TENNV) AS 'Họ và Tên',

DEAN.TENDA,

(PHANCONG.THOIGIAN \* NHANVIEN.LUONG) AS [Tổng tiền lương]

FROM NHANVIEN

JOIN PHANCONG ON NHANVIEN.MANV = PHANCONG.MA\_NV

JOIN DEAN ON PHANCONG.SODA = DEAN.MADA

ORDER BY [Tổng tiền lương] ASC;



#### a. Tạo một View5a dựa trên query này. Bạn sẽ gặp lỗi, tại sao?

CREATE VIEW View5a AS

SELECT CONCAT(NHANVIEN.HONV, ' ', NHANVIEN.TENLOT, ' ', NHANVIEN.TENNV) AS 'Họ và Tên',

DEAN.TENDA,

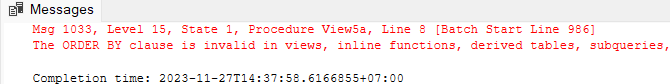
(PHANCONG.THOIGIAN \* NHANVIEN.LUONG) AS [Tổng tiền lương]

FROM NHANVIEN

JOIN PHANCONG ON NHANVIEN.MANV = PHANCONG.MA\_NV

JOIN DEAN ON PHANCONG.SODA = DEAN.MADA

ORDER BY [Tổng tiền lương] ASC;



* Lỗi này là do việc sử dụng câu lệnh ORDER BY trong câu lệnh CREATE VIEW. Trong SQL, một view là một bảng ảo dựa trên kết quả của một câu lệnh SQL. Nó không tồn tại như một bảng thực sự trong cơ sở dữ liệu và không thể được sắp xếp. Có thể sử dụng ORDER BY trong câu lệnh SELECT khi truy vấn một view để sắp xếp kết quả.

#### b. Hãy hiệu chỉnh để có thể tạo được View5b dựa trên query trên.

CREATE VIEW View5b AS

SELECT CONCAT(NHANVIEN.HONV, ' ', NHANVIEN.TENLOT, ' ', NHANVIEN.TENNV) AS 'Họ và Tên',

DEAN.TENDA,

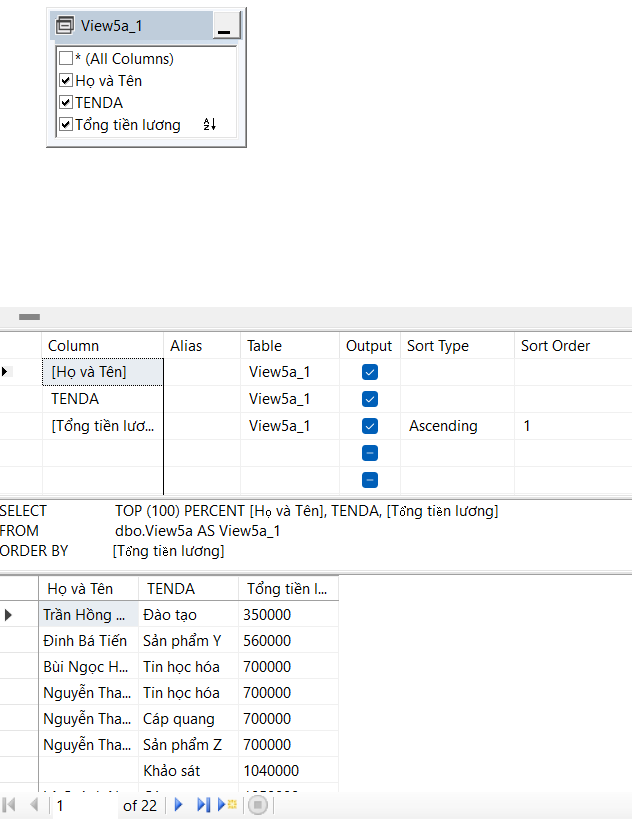
(PHANCONG.THOIGIAN \* NHANVIEN.LUONG) AS [Tổng tiền lương]

FROM NHANVIEN

JOIN PHANCONG ON NHANVIEN.MANV = PHANCONG.MA\_NV

JOIN DEAN ON PHANCONG.SODA = DEAN.MADA

ORDER BY [Tổng tiền lương] ASC;



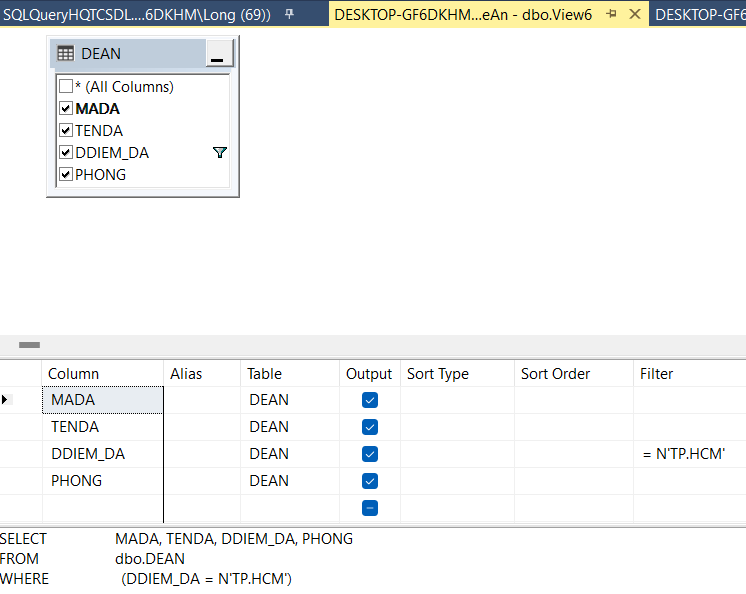
### 6. Tạo View6 gồm các thông tin của bảng DEAN nhưng chỉ lấy các dự án có địa điểm ở Tp.HCM.

CREATE VIEW View6 AS

SELECT \*

FROM DEAN

WHERE DDIEM\_DA = N'TP.HCM';



#### a. Sửa câu View6 thành View6a có thêm tùy chọn WITH ENCRYPTION. Sử dụng sp\_helptext để xem lại mã lệnh. Có xem được không?

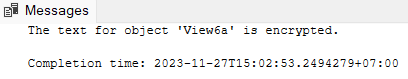
CREATE VIEW View6a WITH ENCRYPTION AS

SELECT \*

FROM DEAN

WHERE DDIEM\_DA = N'TP.HCM';

EXEC sp\_helptext 'View6a';



Mã nguồn của view đã được mã hóa và không thể xem lại nó bằng cách sử dụng sp\_helptext hoặc bất kỳ phương pháp nào khác.

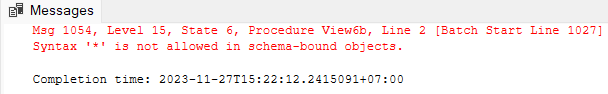
#### b. Sửa câu View6 thành View6b có thêm tùy chọn WITH SCHEMABINDING. Bạn cần hiệu chỉnh điều gì để có thể thực thi được view.

CREATE VIEW View6b WITH SCHEMABINDING AS

SELECT \*

FROM dbo.DEAN

WHERE DDIEM\_DA = N'TP.HCM';



CREATE VIEW View6b

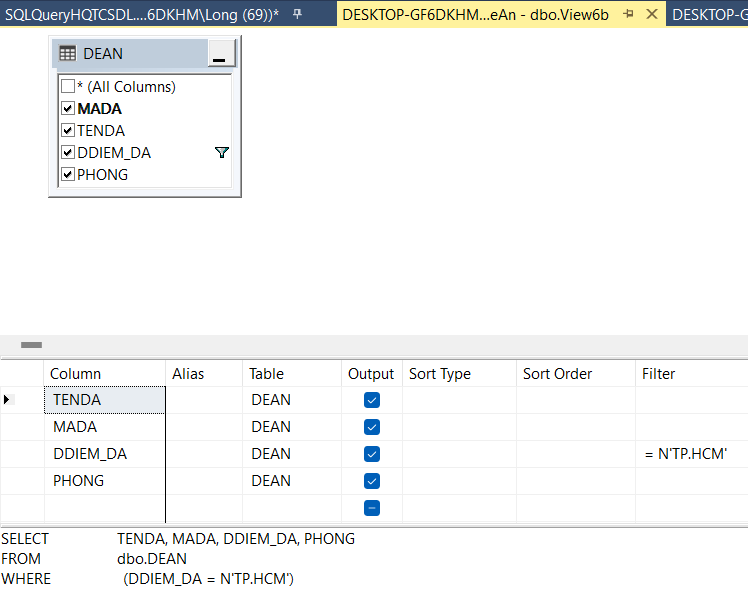
WITH SCHEMABINDING

AS

SELECT TENDA, MADA, DDIEM\_DA, PHONG

FROM dbo.DEAN

WHERE DDIEM\_DA = N'TP.HCM';



Lỗi “Syntax ‘\*’ is not allowed in schema-bound objects” xuất hiện khi ta sử dụng dấu \* trong một view được tạo với tùy chọn WITH SCHEMABINDING. Lí do vì \* biểu thị tất cả các cột và SQL Server không thể xác định rõ ràng các cột nào được sử dụng.

Để khắc phục lỗi thì ta cần chỉ định rõ ràng tên của mỗi cột mà ta muốn lấy từ bảng trong câu lệnh SELECT của view

#### c. Sửa câu View6 thành View6c có thêm tùy chọn WITH CHECK OPTION. Sau đó thêm một dòng dữ liệu sau thông qua View6c:

INSERT View6c

VALUES(‘99’,’Xây dựng đường cao tốc LT-DN’, ‘Đồng Nai’,’5’)

Bạn có thêm được không, tại sao?

Sửa lại dữ liệu để có thể thêm được thông qua View6c.

CREATE VIEW View6c

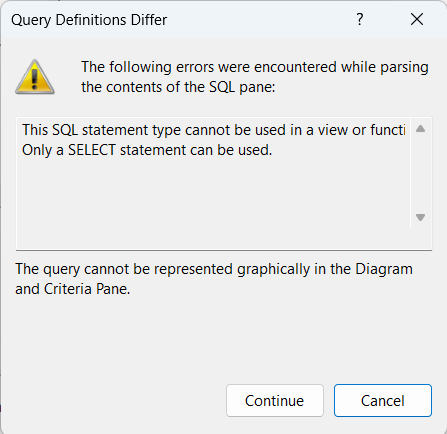
AS

SELECT \*

FROM DEAN

WHERE DDIEM\_DA = N'TP.HCM'

WITH CHECK OPTION;

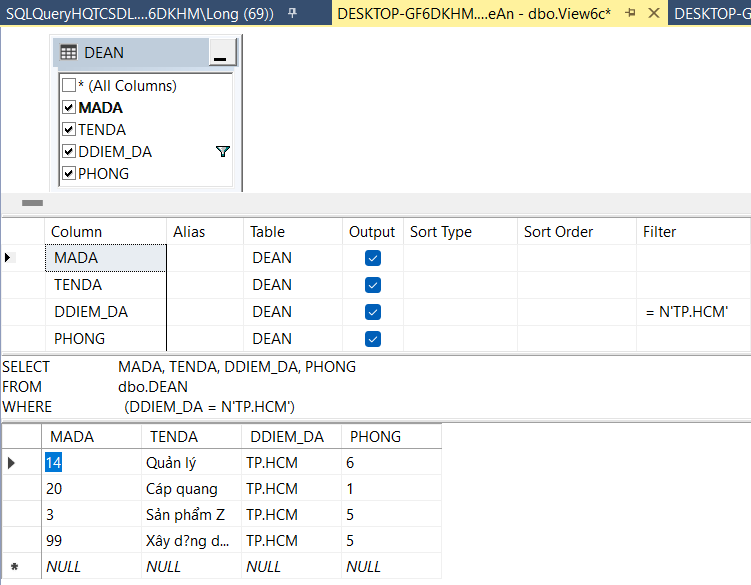


Không thể thêm dòng dữ liệu này thông qua View6c. Lý do là vì View6c đã được tạo với tùy chọn WITH CHECK OPTION để đảm bảo rằng tất cả dữ liệu được thêm hoặc sửa đổi thông qua view phải tuân theo điều kiện WHERE của view. Trong trường hợp này, điều kiện WHERE của View6c là DDIEM\_DA = N'TP.HCM', nhưng dữ liệu đang thêm có DDIEM\_DA = 'Đồng Nai'.

Để thêm dữ liệu thông qua View6c, ta cần sửa lại dữ liệu sao cho ‘DDIEM\_DA’ phải là ‘TP.HCM’.

INSERT INTO View6c

VALUES('99', 'Xây dựng đường cao tốc LT-DN', 'TP.HCM', '5')



#### d. Xóa dữ liệu vừa thêm ở câu c thông qua View6c.

DELETE FROM View6c

WHERE MADA = '99' AND DDIEM\_DA = 'TP.HCM'

### 7. Tạo 3 bảng DEAN\_HCM, DEAN\_HANOI, DEAN\_VT gồm các cột: MaDA varchar(2) Primary Key, TenDA nvarchar(50), DDIEM\_DA varchar(20), PHONG varchar(2). Trong đó :

* DEAN\_HCM: check contraint của DDIEM\_DA là ‘Tp.Hồ Chí Minh’
* DEAN\_HANOI: check contraint của DDIEM\_DA là ‘Hà Nội’
* DEAN\_VT: check contraint của DDIEM\_DA là ‘Vũng Tàu’

Thêm vào mỗi bảng 2 record với ràng buộc tương ứng. Tạo một partition view từ 3 bảng dữ liệu trên. Xem kết quả view vừa tạo được.

-- Tạo bảng DEAN\_HCM

CREATE TABLE DEAN\_HCM (

MaDA varchar(2) PRIMARY KEY,

TenDA nvarchar(50),

DDIEM\_DA varchar(20) CHECK (DDIEM\_DA = 'Tp.Hồ Chí Minh'),

PHONG varchar(2)

);

-- Thêm dữ liệu vào bảng DEAN\_HCM

INSERT INTO DEAN\_HCM VALUES ('01', N'Bảo vệ cây xanh đô thị', N'Tp.Hồ Chí Minh', '01');

INSERT INTO DEAN\_HCM VALUES ('02', N'Giải pháp an toàn thực phẩm', N'Tp.Hồ Chí Minh', '02');

-- Tạo bảng DEAN\_HANOI

CREATE TABLE DEAN\_HANOI (

MaDA varchar(2) PRIMARY KEY,

TenDA nvarchar(50),

DDIEM\_DA varchar(20) CHECK (DDIEM\_DA = 'Hà Nội'),

PHONG varchar(2)

);

-- Thêm dữ liệu vào bảng DEAN\_HANOI

INSERT INTO DEAN\_HANOI VALUES ('03', N'Máy bán cafe tự động', N'Hà Nội', '03');

INSERT INTO DEAN\_HANOI VALUES ('04', N'Phần mềm tổ chức sự kiện', N'Hà Nội', '04');

-- Tạo bảng DEAN\_VT

CREATE TABLE DEAN\_VT (

MaDA varchar(2) PRIMARY KEY,

TenDA nvarchar(50),

DDIEM\_DA varchar(20) CHECK (DDIEM\_DA = 'Vũng Tàu'),

PHONG varchar(2)

);

-- Thêm dữ liệu vào bảng DEAN\_VT

INSERT INTO DEAN\_VT VALUES ('05', N'Nước giải khát thiên nhiên', N'Vũng Tàu', '05');

INSERT INTO DEAN\_VT VALUES ('06', N'Chuỗi nông trại xanh', N'Vũng Tàu', '06');

-- Tạo partition view

CREATE VIEW DEAN\_VIEW AS

SELECT \* FROM DEAN\_HCM

UNION ALL

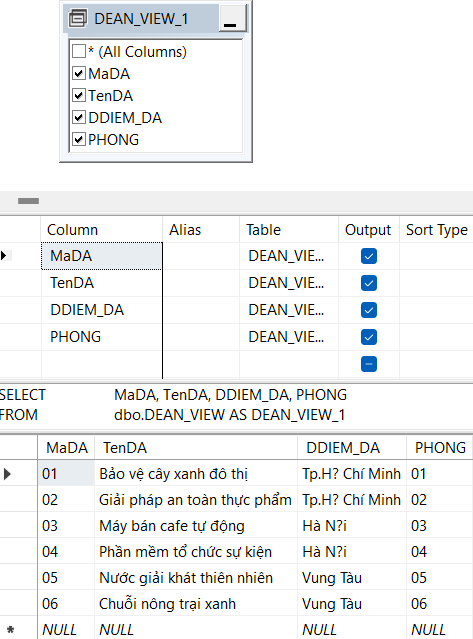
SELECT \* FROM DEAN\_HANOI

UNION ALL

SELECT \* FROM DEAN\_VT;

-- Xem kết quả view

SELECT \* FROM DEAN\_VIEW;



## PHẦN 3: USER-DEFINED FUNCTIONS

### 1. Viết hàm trả về tổng tiền lương trung bình của một phòng ban tùy ý (truyền vào MaPB)

CREATE FUNCTION dbo.hamLuongTrungBinh(@MaPB VARCHAR(2))

RETURNS DECIMAL(18, 2)

AS

BEGIN

DECLARE @LuongTrungBinh DECIMAL(18, 2);

SELECT @LuongTrungBinh = AVG(LUONG)

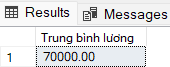
FROM NHANVIEN

WHERE PHG = @MaPB;

RETURN @LuongTrungBinh;

END;

SELECT dbo.hamLuongTrungBinh('2') AS 'Trung bình lương';



### 2. Viết hàm trả về tổng lương nhận được của nhân viên theo dự án (truyền vào MaNV và MaDA)

CREATE FUNCTION dbo.hamTongLuong(@MaNV VARCHAR(3), @MaDA VARCHAR(3))

RETURNS DECIMAL(18, 3)

AS

BEGIN

DECLARE @TongLuong DECIMAL(18, 3);

SELECT @TongLuong = SUM(NHANVIEN.LUONG \* PHANCONG.THOIGIAN)

FROM NHANVIEN

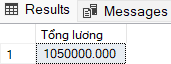
INNER JOIN PHANCONG ON NHANVIEN.MANV = PHANCONG.MA\_NV AND PHANCONG.SODA = @MaDA

WHERE NHANVIEN.MANV = @MaNV;

RETURN @TongLuong;

END;

SELECT dbo.hamTongLuong('001', '20') AS 'Tổng lương';



### 3. Viết hàm trả về tổng tiền lương trung bình của các phòng ban

CREATE FUNCTION dbo.hamTrungBinhPhongBan()

RETURNS TABLE

AS

RETURN (

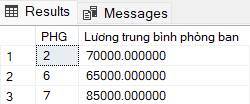
SELECT PHG, AVG(LUONG) AS [Lương trung bình phòng ban]

FROM NHANVIEN

GROUP BY PHG

);

SELECT \* FROM dbo.hamTrungBinhPhongBan();



### 4. Viết hàm trả về tổng tiền thưởng cho nhân viên dựa vào tổng số giờ tham gia dự án(Time\_Total) như sau:

* Nếu Time\_Total >=30 và <=60 thì tổng tiền thưởng = 500 ($)
* Nếu Time\_Total >60 và <100 thì tổng tiền thưởng = 1000 ($)
* Nếu Time\_Total >=100 và <150 thì tổng tiền thưởng =1200($)
* Nếu Time\_Total >=150 thì tổng tiền thưởng = 1600 ($)

CREATE FUNCTION dbo.fn\_TotalBonus(@MaNV VARCHAR(3))

RETURNS DECIMAL(18, 3)

AS

BEGIN

DECLARE @TotalTime INT;

DECLARE @TotalBonus DECIMAL(18, 3);

SELECT @TotalTime = SUM(PHANCONG.THOIGIAN)

FROM PHANCONG

WHERE PHANCONG.MA\_NV = @MaNV;

IF @TotalTime >= 30 AND @TotalTime <= 60

SET @TotalBonus = 500;

ELSE IF @TotalTime > 60 AND @TotalTime < 100

SET @TotalBonus = 1000;

ELSE IF @TotalTime >= 100 AND @TotalTime < 150

SET @TotalBonus = 1200;

ELSE IF @TotalTime >= 150

SET @TotalBonus = 1600;

ELSE

SET @TotalBonus = 0;

RETURN @TotalBonus;

END;

SELECT dbo.fn\_TotalBonus('009') AS 'Tổng tiền thưởng';



### 5. Viết hàm trả ra tổng số dự án theo mỗi phòng ban.

CREATE FUNCTION dbo.hamTongDuAnTheoPhongBan()

RETURNS @TongDuAnTheoPhongBan TABLE

(

Mã\_Phòng\_Ban INT,

Tên\_Phòng\_Ban NVARCHAR(50),

Tổng\_Số\_Dự\_Án INT

)

AS

BEGIN

INSERT INTO @TongDuAnTheoPhongBan (Mã\_Phòng\_Ban, Tên\_Phòng\_Ban, Tổng\_Số\_Dự\_Án)

SELECT PHONGBAN.MAPHG, PHONGBAN.TENPHG, COUNT(DEAN.MADA)

FROM PHONGBAN

LEFT JOIN DEAN

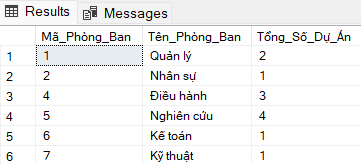
ON PHONGBAN.MAPHG = DEAN.PHONG

GROUP BY PHONGBAN.MAPHG, PHONGBAN.TENPHG;

RETURN;

END;

SELECT \* FROM dbo.hamTongDuAnTheoPhongBan()



### 6. Viết hàm trả về kết quả là một bảng (Table), viết bằng hai cách: Inline Table-Valued Functions và Multistatement Table-Valued. Thông tin gồm: MaNV, HoTen, NgaySinh, NguoiThan, TongLuongTB.

--Inline Table-Valued Functions

CREATE FUNCTION dbo.ThongTinNhanVienInline()

RETURNS TABLE

AS

RETURN

(

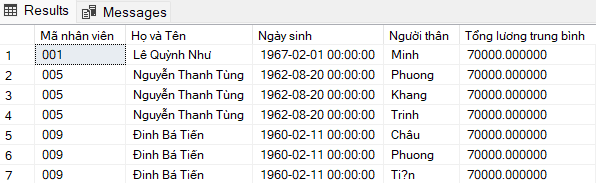
SELECT NHANVIEN.MANV 'Mã nhân viên', CONCAT(NHANVIEN.HONV, ' ', NHANVIEN.TENLOT, ' ', NHANVIEN.TENNV) AS 'Họ và Tên', NHANVIEN.NGSINH 'Ngày sinh', THANNHAN.TENTN AS 'Người thân', AVG(NHANVIEN.LUONG) OVER (PARTITION BY NHANVIEN.MANV) AS 'Tổng lương trung bình'

FROM NHANVIEN

INNER JOIN THANNHAN ON NHANVIEN.MANV = THANNHAN.MA\_NVIEN

);

SELECT \* FROM dbo.ThongTinNhanVienInline();



--Multistatement Table-Valued

CREATE FUNCTION dbo.ThongTinNhanVienMulti()

RETURNS @ThongTinNhanVien TABLE

(

Mã\_Nhân\_Viên NVARCHAR(50),

Họ\_Tên NVARCHAR(100),

Ngày\_Sinh DATE,

Người\_Thân NVARCHAR(100),

Tổng\_Lương\_Trung\_Bình FLOAT

)

AS

BEGIN

INSERT INTO @ThongTinNhanVien (Mã\_Nhân\_Viên, Họ\_Tên, Ngày\_Sinh, Người\_Thân, Tổng\_Lương\_Trung\_Bình)

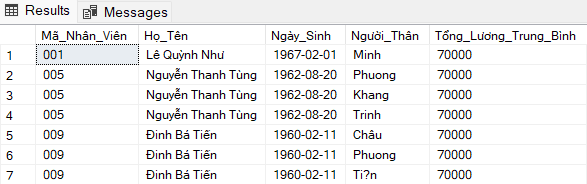
SELECT NHANVIEN.MANV, NHANVIEN.HONV + ' ' + NHANVIEN.TENLOT + ' ' + NHANVIEN.TENNV, NHANVIEN.NGSINH, THANNHAN.TENTN, AVG(NHANVIEN.LUONG) OVER (PARTITION BY NHANVIEN.MANV)

FROM NHANVIEN

INNER JOIN THANNHAN ON NHANVIEN.MANV = THANNHAN.MA\_NVIEN;

RETURN;

END;



# Module 7 & 8

## 1. Việc insert, update, delete, select dữ liệu ở bảng hoàn toàn thực hiện được chỉ bằng câu lệnh DML. Tuy nhiên, trong một số trường hợp, chúng ta cần thiết phải gán quyền cho người dùng (sẽ được đề cập ở module sau) mà không cần thực hiện việc gán quyền trên table thì ta sẽ sử dụng procedure để điều khiển thao tác dữ liệu. Ở bài tập này, yêu cầu sinh viên tạo thủ tục với các thao tác tương ứng như sau:

### a. Tạo thủ tục IN\_PROC\_DEAN cho phép thêm dữ liệu vào bảng đề án với các tham số vào là các field của bảng DEAN

CREATE PROCEDURE IN\_PROC\_DEAN

@TENDA NVARCHAR(50),

@MADA VARCHAR(2),

@DDIEM\_DA VARCHAR(20),

@PHONG VARCHAR(2)

AS

BEGIN

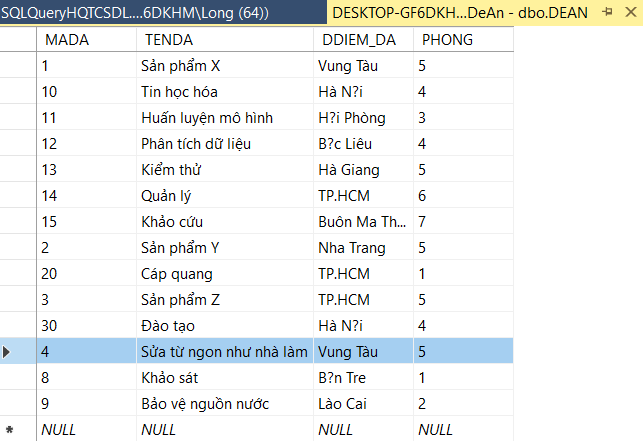
SET NOCOUNT ON;

INSERT INTO DEAN (TENDA, MADA, DDIEM\_DA, PHONG)

VALUES (@TENDA, @MADA, @DDIEM\_DA, @PHONG)

END

EXEC IN\_PROC\_DEAN @MADA = '4', @TENDA = N'Sữa ngon như nhà làm', @DDIEM\_DA = N'Vũng Tàu', @PHONG = '5'



### b. Tạo thủ tục SE\_PRO\_DEAN cho phép xem các đề án có địa điểm đề án được truyền vào khi gọi thủ tục.

CREATE PROCEDURE SE\_PRO\_DEAN

@DDIEM\_DA VARCHAR(20)

AS

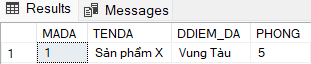
BEGIN

SELECT \* FROM DEAN

WHERE DDIEM\_DA = @DDIEM\_DA

END

EXEC SE\_PRO\_DEAN @DDIEM\_DA = N'Vũng Tàu'



### c. Tạo thủ tục UP\_PROC\_DEAN cho phép cập nhật lại địa điểm đề án là Bà Rịa Vũng Tàu cho các đề án có địa điểm Vung Tau với 2 tham số truyền vào là diadiem\_old, diadiem\_new.

CREATE PROCEDURE UP\_PROC\_DEAN

@diadiem\_old VARCHAR(20),

@diadiem\_new VARCHAR(20)

AS

BEGIN

UPDATE DEAN

SET DDIEM\_DA = @diadiem\_new

WHERE DDIEM\_DA = @diadiem\_old

END

EXEC UP\_PROC\_DEAN @diadiem\_old = N'Vung Tàu', @diadiem\_new = N'Bà Rịa Vũng Tàu'



### d. Tạo thủ tục DEL\_PROC\_DEAN cho phép xóa các đề án đã thêm vào thông qua thủ tục IN\_PROC\_DEAN. Tham số truyền vào là MaDean.

CREATE PROCEDURE DEL\_PROC\_DEAN

@MaDean VARCHAR(2)

AS

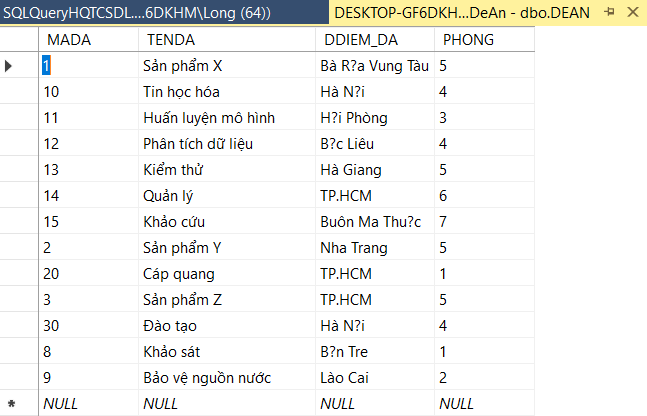
BEGIN

DELETE FROM DEAN

WHERE MADA = @MaDean

END

EXEC DEL\_PROC\_DEAN @MaDean = '4'



## 2. Tạo thủ tục có tham số vào là MaNV, tham số ra (OUTPUT) là tổng số giờ (tổng thời gian) làm việc của nhân viên của tất cả các dự án.

CREATE PROC proc\_name

(@manv datatype, @tong datatype OUTPUT)

AS

BEGIN

--SELECT statement END

--exec

DECLARE @tong datatype

EXEC proc\_name ‘Manv\_data’, @tong OUTPUT

CREATE PROCEDURE TongGioLamViec

@MaNV VARCHAR(9),

@TotalHours INT OUTPUT

AS

BEGIN

SELECT @TotalHours = SUM(THOIGIAN)

FROM PHANCONG

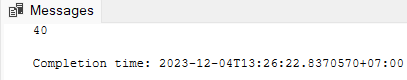
WHERE MA\_NV = @MaNV

END

DECLARE @Hours INT

EXEC TongGioLamViec @MaNV = '009', @TotalHours = @Hours OUTPUT

PRINT @Hours



## 3. Viết thủ tục có tên INSERT\_DEAN để thêm dữ liệu vào bảng DEAN thỏa mãn các ràng buộc sau: kiểm tra MADEAN cần chèn có rỗng không hoặc có trùng với các mã đề án khác đã có trong bảng đề án không. Nếu có hãy thông báo lỗi “Bị trùng mã đề án hoặc mã đề án rỗng, chọn mã đề án khác!!!”. Thực thi thủ tục với 1 trường hợp đúng và 2 trường hợp sai để kiểm chứng.

CREATE PROCEDURE INSERT\_DEAN

@MADA VARCHAR(2),

@TENDA NVARCHAR(50),

@DDIEM\_DA VARCHAR(20),

@PHONG VARCHAR(2)

AS

BEGIN

IF @MADA IS NULL OR EXISTS (SELECT 1 FROM DEAN WHERE MADA = @MADA)

BEGIN

PRINT N'Bị trùng mã đề án hoặc mã đề án rỗng, chọn mã đề án khác!!!'

END

ELSE

BEGIN

INSERT INTO DEAN (MADA, TENDA, DDIEM\_DA, PHONG)

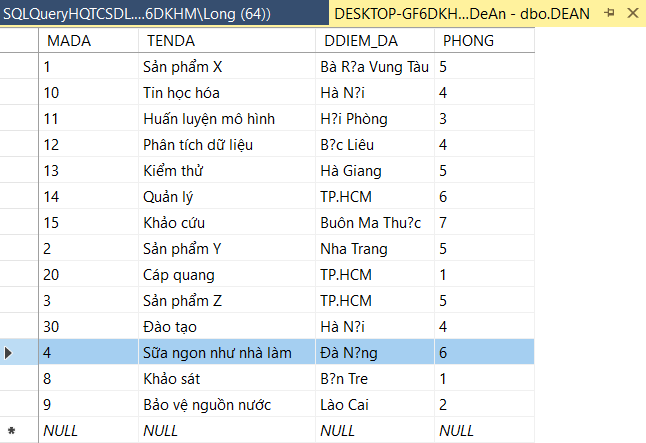
VALUES (@MADA, @TENDA, @DDIEM\_DA, @PHONG)

END

END

--Một trường hợp đúng

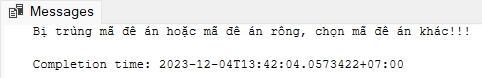
EXEC INSERT\_DEAN @MADA = '4', @TENDA = N'Sữa ngon như nhà làm', @DDIEM\_DA = N'Đà Nẵng', @PHONG = '6'



--Hai trường hợp sai

EXEC INSERT\_DEAN @MADA = '1', @TENDA = N'Gìn giữ cổ thụ nghìn năm', @DDIEM\_DA = N'Đà Nẵng', @PHONG = '6'

EXEC INSERT\_DEAN @MADA = NULL, @TENDA = N'Bảo tồn văn hoá phi vật thể', @DDIEM\_DA = N'Đà Nẵng', @PHONG = '6'



## 4. Viết thủ tục SALARY\_REPORT có sử dụng con trỏ (cursor) để in ra kết quả cửa sổ query analyzer là thông tin giờ dự án cùng với tổng tiền theo dự án của từng nhân viên. (Xem mẫu ví dụ dưới cho 1 nhân viên)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *Nhân viên: 333445555 – Lê Minh Sang* | | | | | | |
| *Dec 8 1945 - Nam* | | | | | | |
| *Mã DA* | *Tên DA* | *Địa điểm* | *Thời gian* | *Lương* | *Tổng* |
| *10* | *Dự án tin học hoá trong QL* | *Ha Noi* | *10* | *4000* | *400000* |
| *20* | *Đường truyền cáp quang* | *Tp HCM* | *10* | *4000* | *400000* |
| *3* | *Đào tạo nhân lực cao cấp* | *Tp HCM* | *10* | *4000* | *400000* |

CREATE PROCEDURE SALARY\_REPORT

AS

BEGIN

DECLARE @MANV VARCHAR(9), @HONV NVARCHAR(15), @TENLOT NVARCHAR(30), @TENNV NVARCHAR(30), @NGSINH SMALLDATETIME, @PHAI NVARCHAR(3), @LUONG NUMERIC(18,0)

DECLARE @MADA VARCHAR(2), @TENDA NVARCHAR(50), @DDIEM\_DA VARCHAR(20), @PHONG VARCHAR(2), @THOIGIAN INT

DECLARE @Total MONEY

DECLARE employee\_cursor CURSOR FOR

SELECT MANV, HONV, TENLOT, TENNV, NGSINH, PHAI, LUONG

FROM NHANVIEN

OPEN employee\_cursor

FETCH NEXT FROM employee\_cursor INTO @MANV, @HONV, @TENLOT, @TENNV, @NGSINH, @PHAI, @LUONG

WHILE @@FETCH\_STATUS = 0

BEGIN

PRINT 'Nhân viên: ' + @MANV + ' – ' + @HONV + ' ' + @TENLOT + ' ' + @TENNV

PRINT CONVERT(VARCHAR, @NGSINH, 107) + ' - ' + @PHAI

PRINT 'Mã DA Tên DA Địa điểm Thời gian Lương Tổng'

DECLARE project\_cursor CURSOR FOR

SELECT MADA, TENDA, DDIEM\_DA, PHONG, THOIGIAN

FROM PHANCONG JOIN DEAN ON PHANCONG.SODA = DEAN.MADA

WHERE MA\_NV = @MANV

OPEN project\_cursor

FETCH NEXT FROM project\_cursor INTO @MADA, @TENDA, @DDIEM\_DA, @PHONG, @THOIGIAN

WHILE @@FETCH\_STATUS = 0

BEGIN

SET @Total = @THOIGIAN \* @LUONG

PRINT @MADA + ' ' + @TENDA + ' ' + @DDIEM\_DA + ' ' + CONVERT(VARCHAR, @THOIGIAN) + ' ' + CONVERT(VARCHAR, @LUONG) + ' ' + CONVERT(VARCHAR, @Total)

FETCH NEXT FROM project\_cursor INTO @MADA, @TENDA, @DDIEM\_DA, @PHONG, @THOIGIAN

END

CLOSE project\_cursor

DEALLOCATE project\_cursor

FETCH NEXT FROM employee\_cursor INTO @MANV, @HONV, @TENLOT, @TENNV, @NGSINH, @PHAI, @LUONG

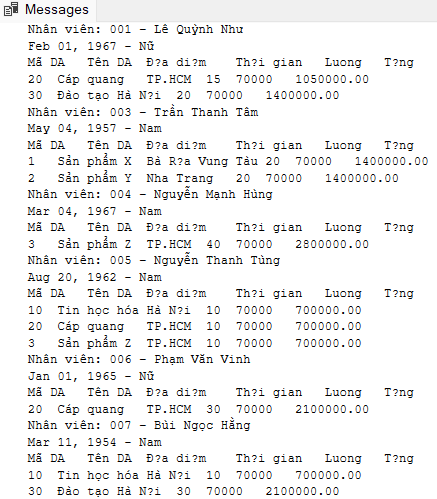
END

CLOSE employee\_cursor

DEALLOCATE employee\_cursor

END

EXEC SALARY\_REPORT



## 5. Viết thủ tục SUM\_SALARY để tính tổng tiền phải trả cho nhân viên với tham số truyền vào là mã nhân viên.

Ví dụ: Tổng tiền phải trả cho nhân viên ‘333445555’ là 400000+400000+400000=1200000.

CREATE PROCEDURE SUM\_SALARY

@MaNV VARCHAR(9),

@TongLuong MONEY OUTPUT

AS

BEGIN

SELECT @TongLuong = SUM(THOIGIAN \* LUONG)

FROM PHANCONG JOIN NHANVIEN ON PHANCONG.MA\_NV = NHANVIEN.MANV

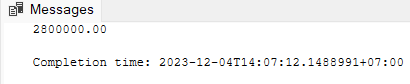
WHERE MA\_NV = @MaNV

END

DECLARE @Luong MONEY

EXEC SUM\_SALARY @MaNV = '009', @TongLuong = @Luong OUTPUT

PRINT @Luong



## 6. Viết thủ tục để kiểm tra việc thêm dữ liệu vào bảng PHANCONG thỏa mãn yêu cầu sau: THOIGIAN phải > 0 và SoDA phải tồn tại ở bảng DeAn. Nếu không thỏa mãn phải thông báo lỗi tương ứng và không được phép insert.

--*hướng dẫn*

*CREATE PROC proc\_name*

*AS*

*BEGIN TRY*

*--do something*

*END TRY*

*BEGIN CATCH*

*ROLLBACK TRANSACTION*

*PRINT ‘Message content’*

*END CATCH*

CREATE PROCEDURE INSERT\_PHANCONG

@MA\_NV VARCHAR(9),

@SODA INT,

@STT INT,

@THOIGIAN INT

AS

BEGIN

IF @THOIGIAN <= 0

BEGIN

PRINT N'Thời gian phải lớn hơn 0!!!'

END

ELSE IF NOT EXISTS (SELECT 1 FROM DEAN WHERE MADA = @SODA)

BEGIN

PRINT N'Số đề án không tồn tại trong bảng Đề án!!!'

END

ELSE

BEGIN

INSERT INTO PHANCONG (MA\_NV, SODA, STT, THOIGIAN)

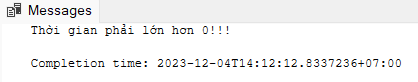
VALUES (@MA\_NV, @SODA, @STT, @THOIGIAN)

END

END

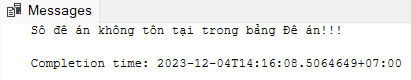
--Giả sử cho thời gian bé hơn 0

EXEC INSERT\_PHANCONG @MA\_NV = '002', @SODA = 1, @STT = 1, @THOIGIAN = -1

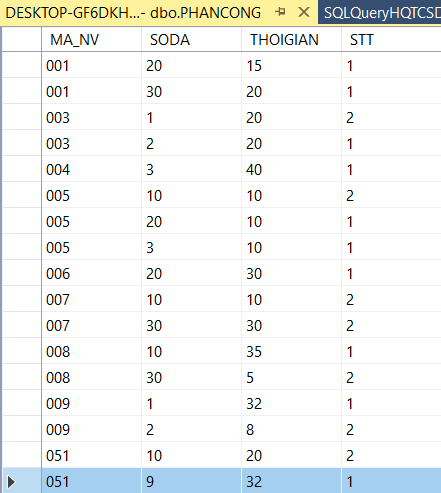


--Giả sử cho số đề án không có trong bảng đề án là 6

EXEC INSERT\_PHANCONG @MA\_NV = '003', @SODA = 6, @STT = 1, @THOIGIAN = 32



EXEC INSERT\_PHANCONG @MA\_NV = '051', @SODA = 9, @STT = 1, @THOIGIAN = 32



## 7. Thêm vào cột HIRE\_DATE (ngày nghỉ hưu) trong bảng nhân viên. Viết thủ tục dùng để cập nhật ngày nghỉ hưu là ngày hiện tại cộng thêm 100 (ngày) cho những nhân viên có tuổi từ 65 trở lên. Tham số truyền vào là field ngaysinh của nhân viên.

--Thêm cột HIRE\_DATE vào bảng NHANVIEN

ALTER TABLE NHANVIEN

ADD HIRE\_DATE DATE

--Tạo một thủ tục lưu trữ để cập nhật ngày nghỉ hưu

CREATE PROCEDURE CapNhatNgayNghiHuu

AS

BEGIN

UPDATE NHANVIEN

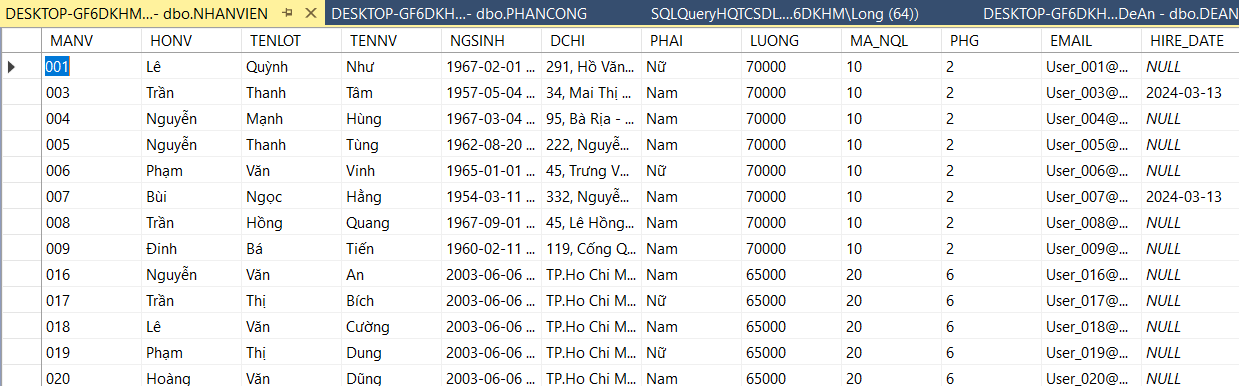
SET HIRE\_DATE = DATEADD(day, 100, GETDATE())

WHERE DATEDIFF(year, NGSINH, GETDATE()) >= 65

END

--Gọi thủ tục

EXEC CapNhatNgayNghiHuu



# Module 9 & 10:

## 1. Tạo triggers trên bảng nhân viên cho thao tác UPDATE. Khi có thao tác UPDATE xảy ra trên field TENNV thì thông báo cho người dùng ‘Không được cập nhật’ và hủy thao tác.

CREATE TRIGGER trg\_Update\_TENNV

ON NHANVIEN

FOR UPDATE

AS

IF UPDATE(TENNV)

BEGIN

RAISERROR ('Không được cập nhật', 16, 1);

ROLLBACK TRANSACTION;

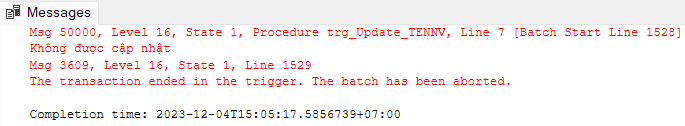
RETURN;

END;

GO

--Kiểm tra trigger

UPDATE NHANVIEN SET TENNV = N'An', PHAI = N'Nam' WHERE MANV = '001';



## 2. Thêm một cột ToTal\_Time vào trong bảng nhân viên. Viết trigger cho thao tác insert, update, delete trên bảng PhanCong. Khi có mẫu tin được thêm vào, hay cập nhật, hay xóa thì ToTal\_Time được tính lại tương ứng cho nhân viên được phân công. Lưu ý, giả sử ban đầu ToTal\_Time = 0. ToTal\_Time là tổng thời gian phân công cho nhân viên của các dự án.

-- Thêm cột ToTal\_Time vào bảng NHANVIEN

ALTER TABLE NHANVIEN

ADD ToTal\_Time INT DEFAULT 0;

-- Tạo trigger cho thao tác INSERT, UPDATE, DELETE trên bảng PHANCONG

CREATE TRIGGER trg\_PhanCong

ON PhanCong

AFTER INSERT, UPDATE, DELETE

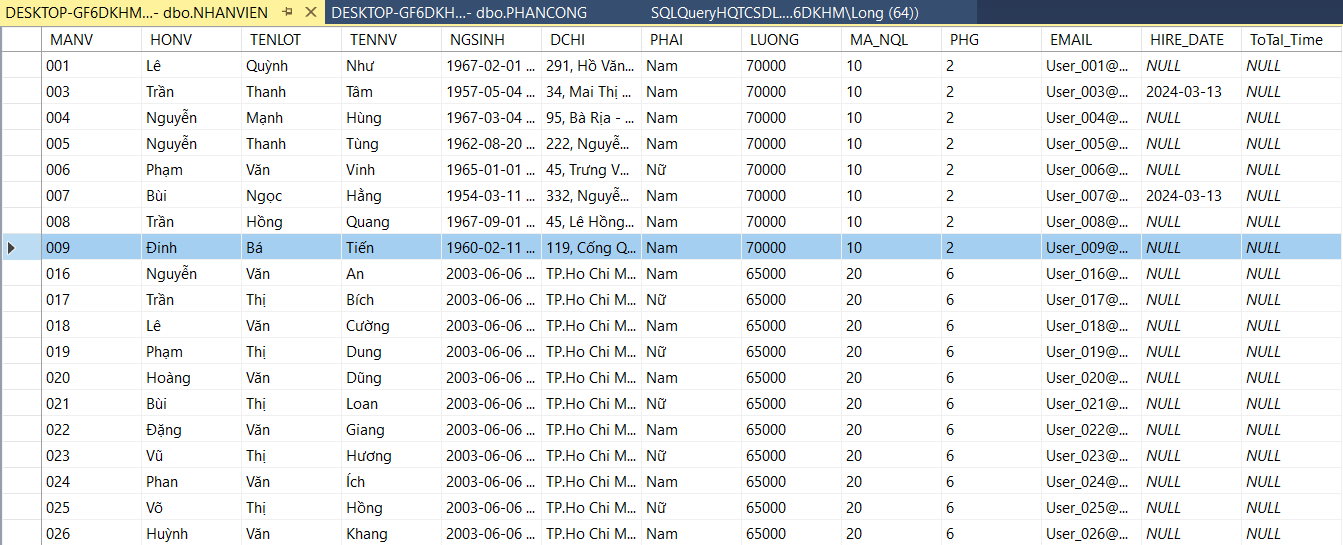
AS

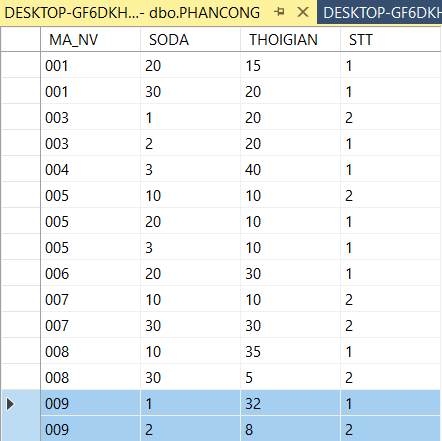
BEGIN

UPDATE NHANVIEN

SET ToTal\_Time = (SELECT SUM(THOIGIAN) FROM PhanCong WHERE MA\_NV = NHANVIEN.MANV)

END

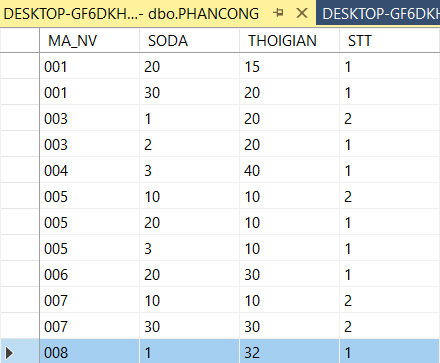


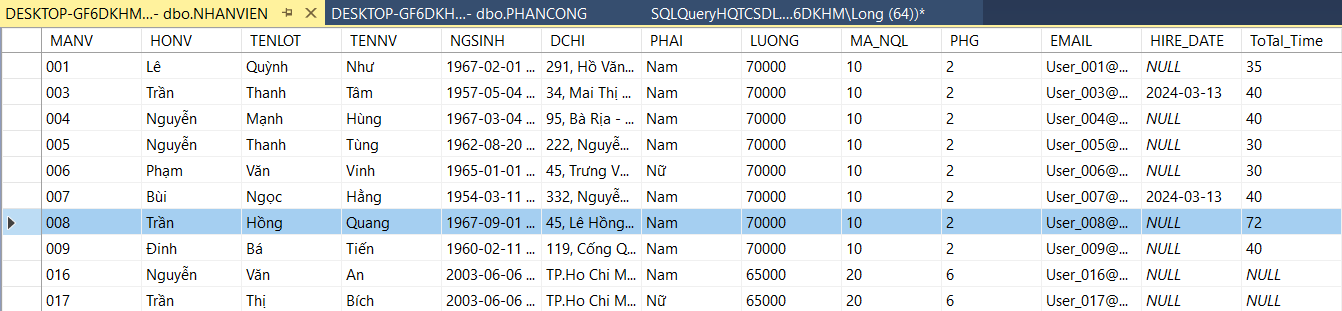


--Thêm một bản ghi vào bảng PhanCong

INSERT INTO PhanCong (MA\_NV, SODA, STT, THOIGIAN)

VALUES ('008', 1, 1, 32)



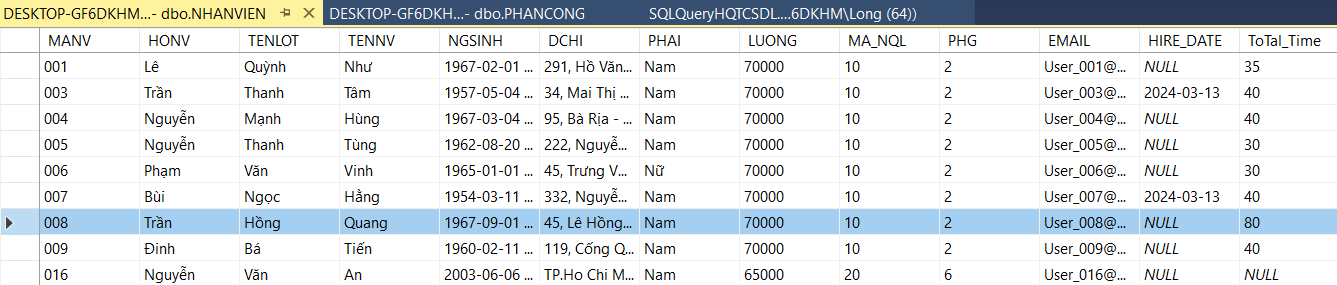


-- Cập nhật một bản ghi trong bảng PhanCong

UPDATE PhanCong

SET THOIGIAN = 40

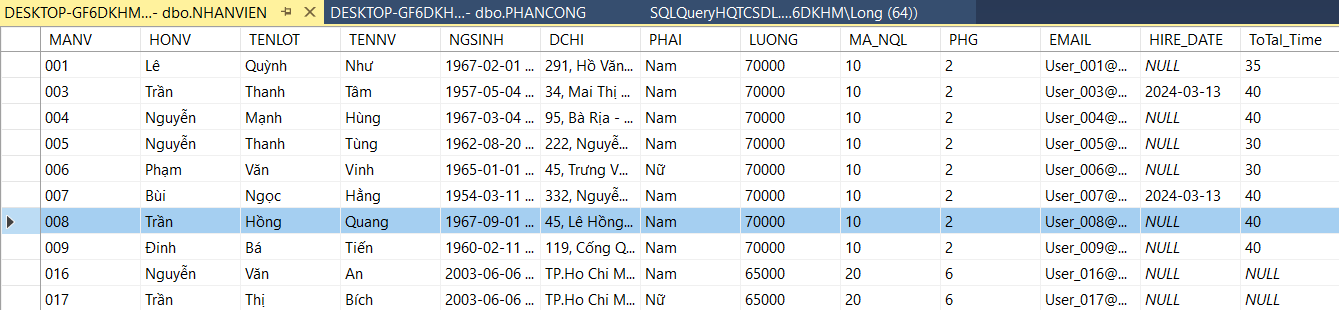
WHERE MA\_NV = '008' AND SODA = 1



-- Xóa một bản ghi từ bảng PhanCong

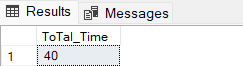
DELETE FROM PhanCong

WHERE MA\_NV = '008' AND SODA = 1



--Kiểm tra

SELECT ToTal\_Time FROM NHANVIEN WHERE MANV = '008'



## 3. Tạo trigger cho thao tác insert, update trên bảng Nhanvien để kiểm tra ràng buộc liên thuộc tính giữa ngày sinh và Hire\_date, trong đó ngày sinh luôn luôn < Hire\_date - 40. (40 là giá trị của datepart ‘yy’).

-- Tạo trigger

CREATE TRIGGER check\_birthdate\_hiredate

ON NHANVIEN

FOR INSERT, UPDATE

AS

BEGIN

-- Kiểm tra ràng buộc liên thuộc tính

IF EXISTS (SELECT \* FROM inserted WHERE DATEDIFF(YEAR, NGSINH, HIRE\_DATE) < 40)

BEGIN

RAISERROR ('Ngày sinh phải nhỏ hơn ngày thuê - 40 năm', 16, 1);

ROLLBACK TRANSACTION;

RETURN;

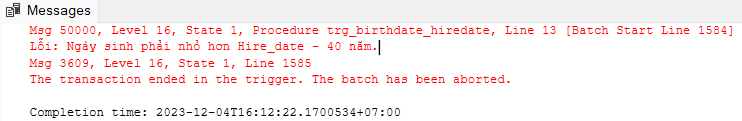
END

END;

-- Thêm một nhân viên mới với ngày sinh và ngày thuê không phù hợp

INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, NGSINH, DCHI, PHAI, LUONG, MA\_NQL, PHG, Hire\_date)

VALUES (N'Nguyễn', N'Quốc', N'Huy', '010', '1980-01-01', N'123, Đường Xóm Đất, TP.HCM', N'Nam', 30000, '005', 5, '2020-01-01');



-- Thêm một nhân viên mới với ngày sinh và ngày thuê phù hợp

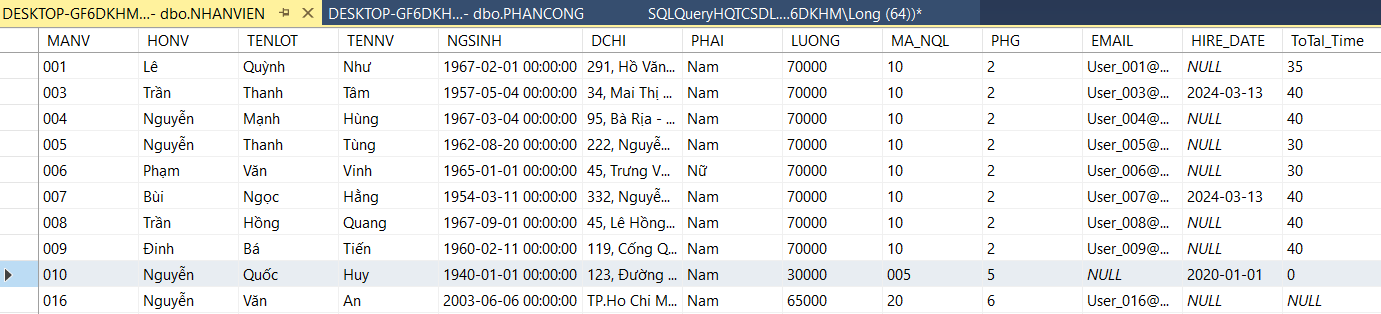
INSERT INTO NHANVIEN (HONV, TENLOT, TENNV, MANV, NGSINH, DCHI, PHAI, LUONG, MA\_NQL, PHG, Hire\_date)

VALUES (N'Nguyễn', N'Quốc', N'Huy', '010', '1980-01-01', N'123, Đường Xóm Đất, TP.HCM', N'Nam', 30000, '005', 5, '2020-01-01');

-- Cập nhật ngày thuê của một nhân viên hiện có

UPDATE NHANVIEN

SET Hire\_date = '2022-01-01'

WHERE MANV = '010';  


## 4. Tạo trigger để kiểm tra thao tác insert trên bảng THANNHAN sao cho số lượng thân nhân được thêm vào tương ứng với một nhân viên không quá 5 người.

-- Tạo trigger

CREATE TRIGGER trg\_insert\_ThanNhan

ON THANNHAN

FOR INSERT

AS

BEGIN

-- Kiểm tra ràng buộc số lượng thân nhân

IF (SELECT COUNT(\*) FROM THANNHAN WHERE MA\_NVIEN IN (SELECT MA\_NVIEN FROM inserted)) > 5

BEGIN

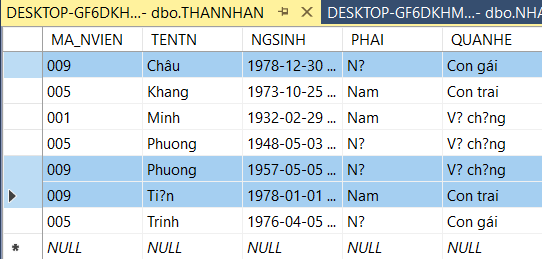
RAISERROR ('Số lượng thân nhân của một nhân viên không được quá 5 người', 16, 1);

ROLLBACK TRANSACTION;

RETURN;

END

END;



Hiện tại, nhân viên mã 009 đang có 3 người thân. Ta tiến hành thêm 2 người thân nữa và người cuối cùng vào là 6 để kiểm tra lỗi

INSERT INTO THANNHAN (MA\_NVIEN, TENTN, PHAI, NGSINH, QUANHE)

VALUES ('009', N'Trinh', N'Nữ', '04/05/1986', N'Con gái')

INSERT INTO THANNHAN (MA\_NVIEN, TENTN, PHAI, NGSINH, QUANHE)

VALUES ('009', N'Tú', N'Nam', '04/05/1996', N'Con trai')

INSERT INTO THANNHAN (MA\_NVIEN, TENTN, PHAI, NGSINH, QUANHE)

VALUES ('009', N'Thiện', N'Nam', '04/05/2006', N'Con trai')

Đến người thứ 6 thì báo lỗi, như vậy trigger đã thực hiện đúng.

