

**DBI202 ASSIGNMENT**

**Database Quản lý cán bộ công chức viên chức TP Hà Nội**

Hoàng Văn Phong

HE187343

Class: SE1913

**I. Giới thiệu**

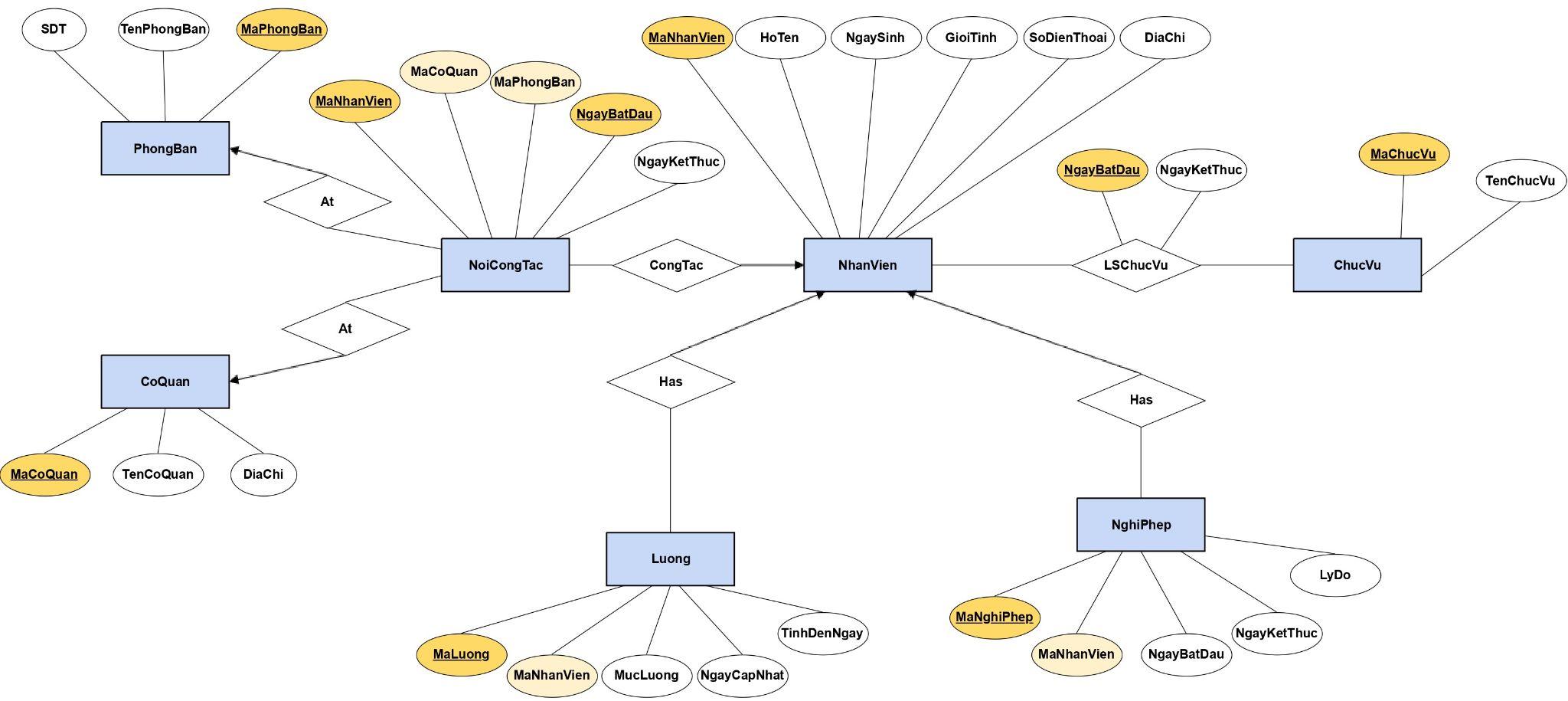
Trong bối cảnh toàn cầu hóa và sự phát triển nhanh chóng của công nghệ thông tin, việc quản lý nguồn nhân lực, đặc biệt là cán bộ công chức viên chức, trở thành một yếu tố quyết định trong công tác quản lý Nhà nước. Thành phố Hà Nội, với vai trò là thủ đô của Việt Nam, không chỉ là trung tâm chính trị, kinh tế mà còn là nơi tập trung nhiều cơ quan, tổ chức và cán bộ công chức viên chức. Do đó, việc xây dựng một hệ thống quản lý hiệu quả cho nguồn nhân lực này là vô cùng cần thiết.

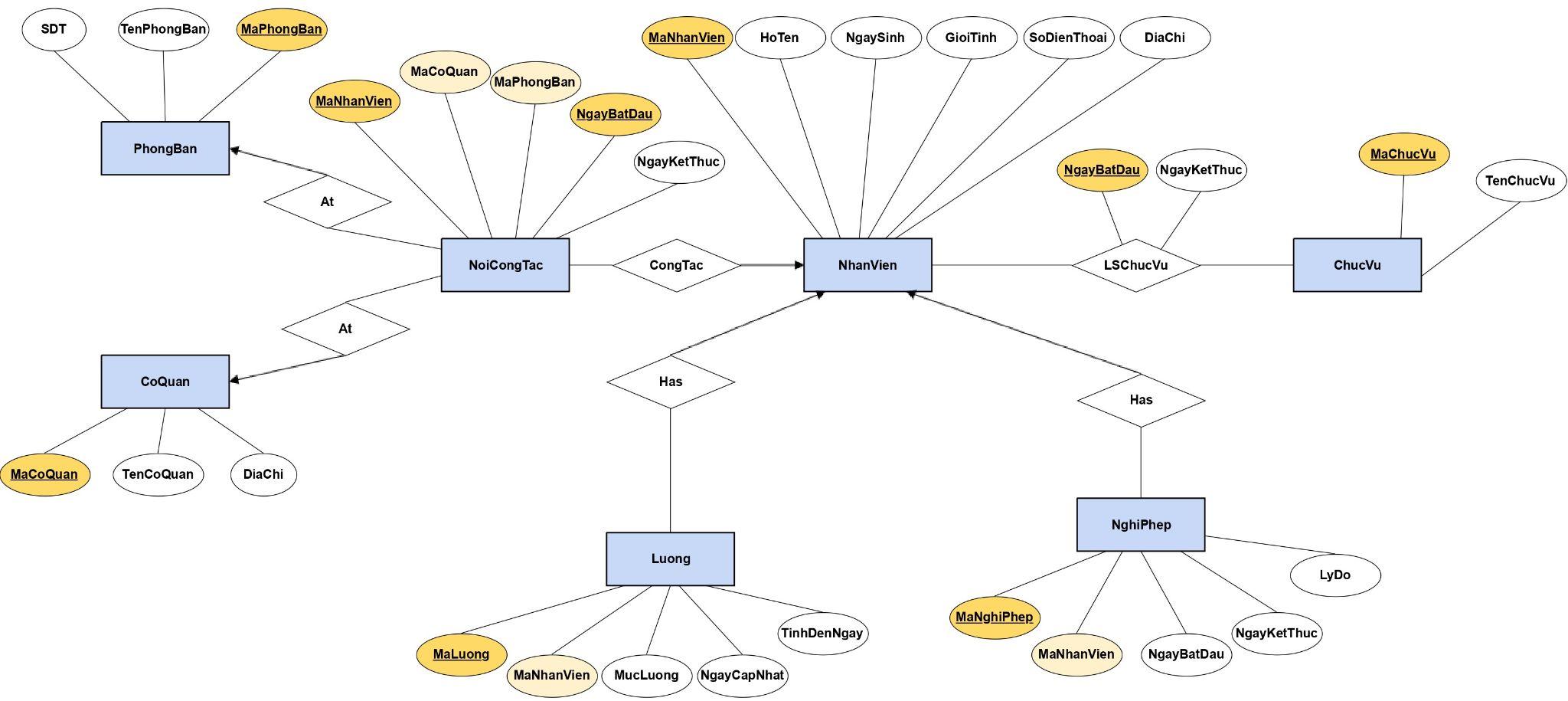
Mục tiêu của dự án này là thiết kế một cơ sở dữ liệu quản lý cán bộ công chức viên chức thành phố Hà Nội, nhằm nâng cao hiệu quả quản lý và phục vụ công tác hành chính công. Cơ sở dữ liệu sẽ cung cấp một nền tảng thông tin chính xác và kịp thời về các cán bộ, công chức viên chức, từ đó hỗ trợ cho việc ra quyết định, quy hoạch nguồn nhân lực và cải cách hành chính.

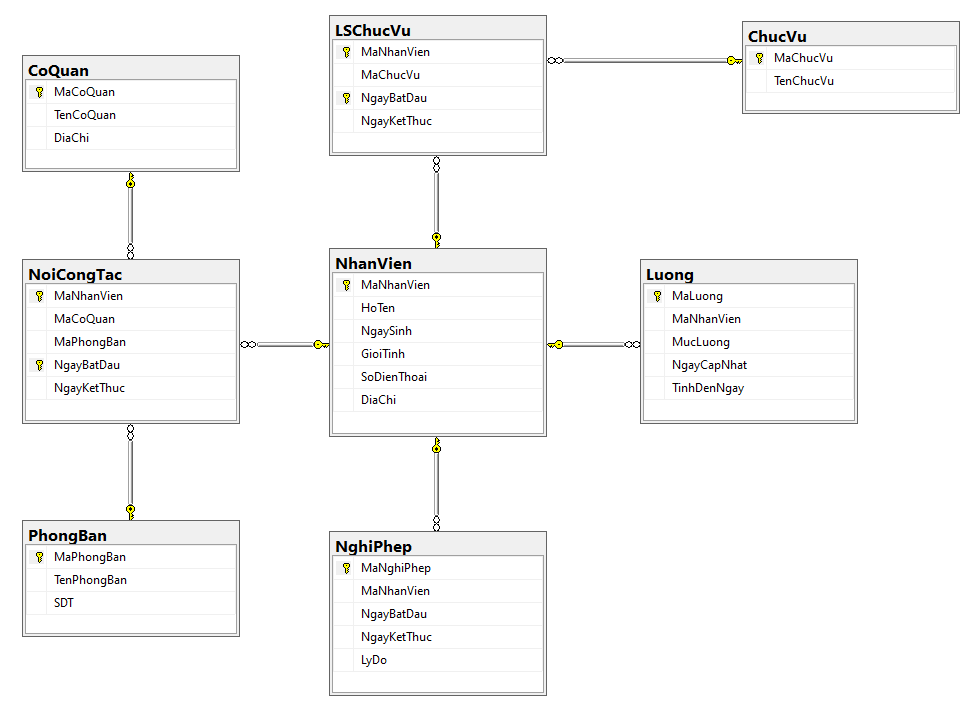
Các công cụ dùng trong bài:

* SQL Server 2019
* SQL Server Management Studio 19
* [Gliffy](https://gliffy.com) : ERD Editor

**II. ERD**





****

**III. Relational Schema**

* **NhanVien**(MaNhanVien, HoTen, NgaySinh, GioiTinh, SoDienThoai, DiaChi)
* **ChucVu**(MaChucVu, TenChucVu)
* **LSChucVu**(MaNhanVien, MaChucVu, NgayBatDau, NgayKetThuc)
* **CoQuan**(MaCoQuan, TenCoQuan, DiaChi)
* **PhongBan**(MaPhongBan, TenPhongBan, SDT)
* **NoiCongTac**(MaNhanVien, MaCoQuan, MaPhongBan, NgayBatDau, NgayKetThuc)
* **Luong**(MaLuong, MaNhanVien, MucLuong, NgayCapNhat, TinhDenNgay)
* **NghiPhep**(MaNghiPhep, MaNhanVien, NgayBatDau, NgayKetThuc, LyDo)

**Chi tiết:**

* Bảng **NhanVien** sẽ lưu trữ thông tin của các nhân viên.
* Bảng **ChucVu** lưu **MaChucVu** và **TenChucVu**.
* Bảng **LSChucVu** lưu lịch sử các chức vụ mà nhân viên đã được nắm giữ, với chức vụ hiện tại của nhân viên thì **NgayKetThuc** sẽ là **NULL**.
* Bảng **CoQuan** và **PhongBan** lưu thông tin của các cơ quan và phòng ban.
* Bảng **NoiCongTac** lưu trữ lịch sử nơi công tác của nhân viên, với nơi làm việc hiện tại thì **NgayKetThuc** sẽ là **NULL**. **MaPhongBan** có thế là **NULL** nếu nhân viên đó có chức vụ cao.
* Bảng **Luong** lưu thông tin lương của nhân viên, với mức lương hiện tại của nhân viên thì **TinhDenNgay** sẽ là **NULL**.
* Bảng **NghiPhep** lưu thông tin nghỉ phép của nhân viên.

**IV. SQL commands**

**1. Tạo database, tạo bảng**

Trước khi tạo database, ta sẽ kiểm tra xem database đã tồn tại chưa, nếu đã tồn tại thì sẽ DROP DATABASE:

IF EXISTS(SELECT name FROM sys.databases

WHERE name = 'DBI202\_Assignment\_PhongHV')

DROP DATABASE DBI202\_Assignment\_PhongHV

Sau đó sẽ tạo database mới:

CREATE DATABASE DBI202\_Assignment\_PhongHV

Sau khi đã tạo database thì tiếp theo sẽ tạo các bảng. Trước khi tạo bảng, phải chọn database ta muốn tạo bảng:

USE DBI202\_Assignment\_PhongHV

Tạo các bảng:

CREATE TABLE NhanVien(

MaNhanVien INT NOT NULL PRIMARY KEY,

HoTen NVARCHAR(50) NOT NULL,

NgaySinh DATE,

GioiTinh BIT,

SoDienThoai NVARCHAR(10),

DiaChi NVARCHAR(255)

)

CREATE TABLE CoQuan(

MaCoQuan INT NOT NULL PRIMARY KEY,

TenCoQuan NVARCHAR(100) NOT NULL,

DiaChi NVARCHAR(255)

)

CREATE TABLE PhongBan(

MaPhongBan INT NOT NULL PRIMARY KEY,

TenPhongBan NVARCHAR(100) NOT NULL,

SDT NVARCHAR(10)

)

CREATE TABLE ChucVu(

MaChucVu INT NOT NULL PRIMARY KEY,

TenChucVu NVARCHAR(50) NOT NULL

)

CREATE TABLE NoiCongTac(

MaNhanVien INT NOT NULL,

MaCoQuan INT NOT NULL,

MaPhongBan INT NULL,

NgayBatDau DATE NOT NULL,

NgayKetThuc DATE NULL,

PRIMARY KEY (MaNhanVien, NgayBatDau),

CONSTRAINT FK\_NV\_CT FOREIGN KEY (MaNhanVien) REFERENCES NhanVien(MaNhanVien),

CONSTRAINT FK\_PB\_CT FOREIGN KEY (MaPhongBan) REFERENCES PhongBan(MaPhongBan),

CONSTRAINT FK\_CQ\_CT FOREIGN KEY (MaCoQuan) REFERENCES CoQuan(MaCoQuan)

)

CREATE TABLE LSChucVu(

MaNhanVien INT NOT NULL,

MaChucVu INT NOT NULL,

NgayBatDau DATE NOT NULL,

NgayKetThuc DATE NULL,

PRIMARY KEY (MaNhanVien, NgayBatDau),

CONSTRAINT FK\_NV\_LSCV FOREIGN KEY (MaNhanVien) REFERENCES NhanVien(MaNhanVien),

CONSTRAINT FK\_CV\_LSCV FOREIGN KEY (MaChucVu) REFERENCES ChucVu(MaChucVu)

)

CREATE TABLE Luong(

MaLuong INT IDENTITY(1,1) NOT NULL PRIMARY KEY,

MaNhanVien INT NOT NULL,

MucLuong FLOAT NOT NULL,

NgayCapNhat DATE NOT NULL,

TinhDenNgay DATE NULL,

CONSTRAINT FK\_NV\_Luong FOREIGN KEY (MaNhanVien) REFERENCES NhanVien(MaNhanVien)

)

CREATE TABLE NghiPhep(

MaNghiPhep INT NOT NULL PRIMARY KEY,

MaNhanVien INT NOT NULL,

NgayBatDau DATE NOT NULL,

NgayKetThuc DATE NOT NULL,

LyDo NVARCHAR(255) NULL,

CONSTRAINT FK\_NV\_NghiPhep FOREIGN KEY (MaNhanVien) REFERENCES NhanVien(MaNhanVien)

)

**2. Tạo role và cấp quyền cho người dùng**

Sau khi đã tạo các bảng, bước tiếp theo là tạo các role và cấp quyền cho từng role:

CREATE ROLE [admin]

CREATE ROLE viewer

CREATE ROLE data\_manager

GRANT CONTROL ON DATABASE::DBI202\_Assignment\_PhongHV TO [admin]

GRANT SELECT ON DATABASE::DBI202\_Assignment\_PhongHV TO viewer

GRANT SELECT, INSERT, UPDATE, DELETE ON DATABASE::DBI202\_Assignment\_PhongHV TO data\_manager

Với admin sẽ có mọi quyền trên database, Viewer sẽ chỉ có thể xem dữ liệu trên database, Data Manager sẽ có quyền xem, sửa, xóa và cập nhật dữ liệu trên database.

Tạo các user vào tương ứng với các role:

CREATE LOGIN DBadmin WITH PASSWORD = 'admin'

CREATE USER DBadmin FOR LOGIN DBadmin

ALTER ROLE [admin] ADD MEMBER DBadmin

CREATE LOGIN DBviewer WITH PASSWORD = '123'

CREATE USER DBviewer FOR LOGIN DBviewer

ALTER ROLE viewer ADD MEMBER DBviewer

CREATE LOGIN DTmanager WITH PASSWORD = '123'

CREATE USER DTmanager FOR LOGIN DTmanager

ALTER ROLE data\_manager ADD MEMBER DTmanager

**3. Tạo các view**

GO

CREATE VIEW ChucVuHienTai AS

SELECT n.MaNhanVien, n.HoTen, c.TenChucVu

FROM NhanVien n JOIN LSChucVu l on n.MaNhanVien = l.MaNhanVien

JOIN ChucVu c on l.MaChucVu=c.MaChucVu

WHERE l.NgayKetThuc IS NULL;

GO

CREATE VIEW NoiCongTacHienTai AS

SELECT n.MaNhanVien, n.HoTen, q.TenCoQuan, p.TenPhongBan

FROM NhanVien n JOIN NoiCongTac ct on n.MaNhanVien=ct.MaNhanVien

JOIN CoQuan q on q.MaCoQuan=ct.MaCoQuan

LEFT JOIN PhongBan p on p.MaPhongBan=ct.MaPhongBan

WHERE ct.NgayKetThuc IS NULL;

**4. Tạo các procedure**

GO

CREATE PROC NhanVienTaiPhongBan

@PhongBan NVARCHAR(100)

AS BEGIN

SELECT n.MaNhanVien, n.HoTen, n.GioiTinh, n.SoDienThoai, n.DiaChi

FROM NhanVien n JOIN NoiCongTac c on n.MaNhanVien=c.MaNhanVien

JOIN PhongBan p on p.MaPhongBan=c.MaPhongBan

WHERE p.TenPhongBan LIKE @PhongBan AND c.NgayKetThuc IS NULL

END;

GO

CREATE PROC NhanVienTaiCoQuan

@CoQuan NVARCHAR(100)

AS BEGIN

SELECT n.MaNhanVien, n.HoTen, n.GioiTinh, n.SoDienThoai, n.DiaChi

FROM NhanVien n JOIN NoiCongTac c on n.MaNhanVien=c.MaNhanVien

JOIN CoQuan q on q.MaCoQuan=c.MaCoQuan

WHERE q.TenCoQuan LIKE @CoQuan AND c.NgayKetThuc IS NULL

END;

**5. Tạo các function**

GO

CREATE FUNCTION SoNhanVienPhongBan(@PhongBan NVARCHAR(100))

RETURNS INT

AS BEGIN

DECLARE @Count INT;

SELECT @Count = COUNT(n.MaNhanVien)

FROM NoiCongTac n JOIN PhongBan p on n.MaPhongBan = p.MaPhongBan

WHERE P.TenPhongBan LIKE @PhongBan AND n.NgayKetThuc IS NULL

GROUP BY p.MaPhongBan

RETURN @Count

END;

GO

CREATE FUNCTION SoNhanVienCoQuan(@CoQuan NVARCHAR(100))

RETURNS INT

AS BEGIN

DECLARE @Count INT;

SELECT @Count = COUNT(n.MaNhanVien)

FROM NoiCongTac n JOIN CoQuan q on n.MaCoQuan = q.MaCoQuan

WHERE q.TenCoQuan LIKE @CoQuan AND n.NgayKetThuc IS NULL

GROUP BY q.MaCoQuan

RETURN @Count

END;

**6. Tạo các trigger**

GO

CREATE TRIGGER CapNhatLuong

ON Luong

INSTEAD OF INSERT

AS BEGIN

DECLARE @MaNV int, @NgayCN DATE, @ML FLOAT, @NgayKT DATE

SELECT @MaNV = MaNhanVien, @NgayCN = NgayCapNhat, @ML = MucLuong, @NgayKT = TinhDenNgay FROM inserted

IF EXISTS (SELECT \* FROM Luong WHERE MaNhanVien = @MaNV AND TinhDenNgay IS NULL) BEGIN

IF (@NgayCN > (SELECT TOP 1 NgayCapNhat FROM Luong WHERE MaNhanVien = @MaNV AND TinhDenNgay IS NULL)) BEGIN

UPDATE Luong SET TinhDenNgay = @NgayCN WHERE TinhDenNgay IS NULL AND MaNhanVien = @MaNV

INSERT INTO Luong(MaNhanVien, MucLuong, NgayCapNhat, TinhDenNgay) VALUES

(@MaNV, @ML, @NgayCN, @NgayKT)

END

ELSE

PRINT 'Ngay cap nhat khong hop le'

END

ELSE

INSERT INTO Luong(MaNhanVien, MucLuong, NgayCapNhat, TinhDenNgay) VALUES

(@MaNV, @ML, @NgayCN, @NgayKT)

END;

GO

CREATE TRIGGER KiemTraSDTNhanVien

ON NhanVien

AFTER INSERT

AS

BEGIN

IF EXISTS (SELECT \* FROM inserted

WHERE SoDienThoai NOT LIKE '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'

)BEGIN

PRINT 'So dien thoai khong hop le';

ROLLBACK TRANSACTION;

END

END;

GO

CREATE TRIGGER KiemTraSDTPhongBan

ON PhongBan

AFTER INSERT

AS

BEGIN

IF EXISTS (SELECT \* FROM inserted

WHERE SDT NOT LIKE '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'

)BEGIN

PRINT 'So dien thoai khong hop le';

ROLLBACK TRANSACTION;

END

END;

**7. Select**

Hiển thị toàn bộ nhân viên sắp xếp theo tên:

SELECT \*

FROM NhanVien

ORDER BY HoTen

Hiển thị những nhân viên làm việc tại Sở Y Tế:

SELECT n.MaNhanVien, n.HoTen, n.GioiTinh, n.NgaySinh, n.SoDienThoai, n.DiaChi

FROM NhanVien n JOIN NoiCongTac c ON n.MaNhanVien = c.MaNhanVien

LEFT JOIN CoQuan q ON c.MaCoQuan = q.MaCoQuan

WHERE q.TenCoQuan = N'Sở Y Tế' AND c.NgayKetThuc IS NULL;

**V. SQL files**

1. [DBScript.sql](https://github.com/phongphong1/DBI202_Assignment/blob/main/DBScript.sql): Bao gồm các lệnh tạo DB, tạo bảng, cấp quyền người dùng và insert dữ liệu.

2. [Helper.sql](https://github.com/phongphong1/DBI202_Assignment/blob/main/Helper.sql): Bao gồm các lệnh tạo View, Procedure, Function và Trigger.

3. [Select.sql](https://github.com/phongphong1/DBI202_Assignment/blob/main/Select.sql): Bao gồm các lệnh select.