

ĐẠI HỌC THỦY LỢI  
KHOA CÔNG NGHỆ THÔNG TIN



11

BÁO CÁO BÀI TẬP LỚN  
HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

Đề tài: Quản lý rạp chiếu phim

*Giáo viên hướng dẫn: Nguyễn Đức Huy*

*Sinh viên thực hiện:*

*Lê Minh Thành                      59Th2*

*Nghiêm Minh Quý                59Th2*

*Trần Xuân Hình                    59Th2*



## BÀI TẬP LỚN

### HỌC PHẦN: HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

Mã đề tài: 14

1. Tên đề tài

Xây dựng cơ sở dữ liệu cho bài toán quản lý hoạt động của rạp chiếu phim.

2. Mục đích

Xây dựng cơ sở dữ liệu cho bài toán trên hệ quản trị MS SQL Server.

3. Công việc cần thực hiện

- Tìm hiểu, phân tích yêu cầu bài toán (Đã nêu trong phụ lục danh sách bài tập lớn)
- Xây dựng mô hình thực thể - liên kết
- Chuyển mô hình thực thể-liên kết sang mô hình dữ liệu quan hệ
- Cài đặt cơ sở dữ liệu trên hệ quản trị MS SQL Server
- Phân chia công việc của nhóm và thực hiện theo đúng tiến độ đề ra
- Làm báo cáo bài tập lớn
- Bảo vệ bài tập lớn

4. Yêu cầu

- Kết quả làm bài tập lớn: Báo cáo bài tập lớn + Cơ sở dữ liệu

# Mục lục

HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU	1
<b>BÀI TẬP LỚN</b>	<b>2</b>
Phân tích thiết kế hệ thống	4
Mô tả bài toán	4
<b>Chương 1: Xây dựng mô hình thực thể - liên kết</b>	<b>6</b>
<b>Chương 2: Xây dựng mô hình dữ liệu quan hệ</b>	<b>8</b>
Xác định kiểu dữ liệu và các ràng buộc cần thiết trên các quan hệ	8
Mô hình quan hệ ERD	14
<b>Chương 3: Cài đặt cơ sở dữ liệu</b>	<b>16</b>
Tạo cơ sở dữ liệu	16
Tạo các bảng và ràng buộc dữ liệu trên bảng	17
Tạo các view	22
Tạo các Store Procedure	25
Tạo các Function	34

# Phân tích thiết kế hệ thống

## 1. Mô tả bài toán

- Quản lý thông tin của các nhân viên: Họ tên, Ngày sinh, Quê quán, Số CMT, Giới tính, Chức vụ
- Hệ thống quản lý thông tin phim bao gồm: Mã phim, Tên phim, Đạo diễn, Diễn viên, Thể loại, Thời lượng, Ngày khởi chiếu, Xếp loại, Mô tả, Định dạng, Ngôn ngữ, Nhà sản xuất
- Quản lý phòng chiếu phim bao gồm: Mã phòng, Tên phòng, Số ghế thường, Số ghế VIP
- Quản lý lịch chiếu phim bao gồm: Mã phim, Định dạng phim, Ngôn ngữ phim, Mã phòng, Thời gian chiếu
- Quản lý đơn giá vé bao gồm: Định dạng phim, Thời gian theo thứ trong tuần, Thời gian các suất chiếu, Đơn giá
- Quản lý thông tin của các khách hàng: Họ tên, Ngày sinh, Giới tính
- Quản lý thông tin tài khoản của các nhân viên: Tên đăng nhập, Mật khẩu
- Quản lý số lượng vé bán được của mỗi phim theo lịch chiếu: Mã
- Việc bán vé của hệ thống sẽ được thực hiện trực tiếp tại rạp hoặc qua ứng dụng, cụ thể như sau: Khi 1 khách hàng muốn mua vé, họ có thể đến trực tiếp rạp để mua vé hoặc có thể đặt vé qua ứng dụng.
  - + Khách hàng đến mua vé trực tiếp tại rạp: Khách hàng lựa chọn phim và đặt vé với nhân viên của rạp
  - + Khách hàng mua vé qua ứng dụng: Khách hàng lựa chọn phim và xem chi tiết thông tin của bộ phim cũng như lịch chiếu của phim đó để đặt vé và trả tiền thông qua phương thức chuyển khoản



# Chương 1: Xây dựng mô hình thực thể - liên kết

## 1. Xác định các thực thể và thuộc tính

- Thực thể: Phim

Thuộc tính: IDPhim, ID\_NSX, TenPhim, Poster, ThoiLuong, KhoiChieu, Rated, MoTa

- Thực thể: PhongChieuPhim

Thuộc tính: IDPhong, TenPhong, SoGheThuong , SoGheVIP

- Thực thể: NhaSanXuat

Thuộc tính: ID\_NSX, TenNSX

- Thực thể: TheLoai

Thuộc tính: IDTheLoai, TenTheLoai

- Thực thể: DinhDang

Thuộc tính: ID\_DinhDang, TenDinhDang

- Thực thể: NgonNgu

Thuộc tính: ID\_NgonNgu, NgonNgu

- Thực thể: DaoDien

Thuộc tính: IDDaoDien, TenDaoDien

- Thực thể: DienVien

Thuộc tính: IDDienVien, TenDienVien

- Thực thể: P\_DaoDien

Thuộc tính: IDDienVien, TenDienVien

- Thực thể: P\_DienVien

Thuộc tính: IDPhim, IDDienVien

- Thực thể: P\_TheLoai

Thuộc tính: IDPhim, IDTheLoai

- Thực thể: P\_DN

Thuộc tính: ID\_PD\_N, IDPhim, ID\_DinhDang, ID\_NgonNgu

- Thực thể: LichChieuPhim

Thuộc tính: ID\_LichChieu, ID\_PD\_N, IDPhong, ThoiGianChieu

- Thực thể: Account\_KH

Thuộc tính: IDKhachHang, TenDangNhap, MatKhau

- Thực thể: KhachHang

Thuộc tính: IDKhachHang, HoTen, NgaySinh, GioiTinh

- Thực thể: NhanVien  
Thuộc tính: IDNhanVien, HoTen, NgaySinh, Que, SoChungMinhThu, GioiTinh
- Thực thể: ChucVu  
Thuộc tính: IDChucVu, ChucVu
- Thực thể: ThoiGian  
Thuộc tính: ID\_TG, TG\_BatDau, TG\_KetThuc
- Thực thể: DonGia  
Thuộc tính: ID\_DinhDang, Thu, ID\_TG, DonGia
- Thực thể: Booking\_POS  
Thuộc tính: ID, ID\_LichChieu, TG\_MuaVe, SoLuongVe, MaGheNgoi, GheVIP, TongTien
- Thực thể: Booking\_Client  
Thuộc tính: ID, IDKhachHang, ID\_LichChieu, TG\_MuaVe, SoLuongVe, MaGheNgoi, GheVIP, TongTien

## Chương 2: Xây dựng mô hình dữ liệu quan hệ

### 1. Chuyển đổi từ mô hình thực thể -liên kết sang mô hình quan hệ

- Phim: (IDPhim, ID\_NSX, TenPhim, Poster, ThoiLuong, KhoiChieu, Rated, MoTa)
- PhongChieuPhim: (IDPhong, TenPhong, SoGheThuong , SoGheVIP)
- NhaSanXuat: (ID\_NSX, TenNSX)
- TheLoai: (IDTheLoai, TenTheLoai)
- DinhDang: (ID\_DinhDang, TenDinhDang)
- NgonNgu: (ID\_NgonNgu, NgonNgu)
- DaoDien: (IDDaoDien, TenDaoDien)
- DienVien: (IDDienVien, IDDaoDien)
- P\_DaoDien: (IDPhim, TenDienVien)
- P\_DienVien: (IDPhim, IDDienVien)
- P\_TheLoai: (IDPhim, IDTheLoai)
- P\_DN: (ID\_PDN, IDPhim, ID\_DinhDang, ID\_NgonNgu)
- LichChieuPhim: (ID\_LichChieu, ID\_PDN, IDPhong, ThoiGianChieu)
- Account\_KH: (IDKhachHang, TenDangNhap, MatKhau)
- KhachHang: (IDKhachHang, HoTen, NgaySinh, GioiTinh)
- NhanVien: (IDNhanVien, HoTen, NgaySinh, Que, SoChungMinhThu, GioiTinh)
- ChucVu: (IDChucVu, ChucVu)
- ThoiGian: (ID\_TG, TG\_BatDau, TG\_KetThuc)
- DonGia: (ID\_DinhDang, Thu, ID\_TG, DonGia)
- Booking\_POS: (ID, ID\_LichChieu, TG\_MuaVe, SoLuongVe, MaGheNgoi, GheVIP, TongTien)
- Booking\_Client: (ID, IDKhachHang, ID\_LichChieu, TG\_MuaVe, SoLuongVe, MaGheNgoi, GheVIP, TongTien)

### 2. Xác định kiểu dữ liệu và các ràng buộc cần thiết trên các quan hệ

#### - Quan hệ **Phim**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDPhim	int		PK	



ID_NSX	int			
TenPhim	nvarchar	NVARCHAR(MAX)		
Poster	image			
ThoiLuong	tinyint			
KhoiChieu	date			
Rated	char	3		
MoTa	nvarchar	NVARCHAR(MAX)		

- Quan hệ **PhongChieuPhim**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDPhong	tinyint		PK	
TenPhong	varchar	5		
SoGheThuong	tinyint			
SoGheVIP	tinyint			

- Quan hệ **NhaSanXuat**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
ID_NSX	int			
TenNSX	nvarchar	20		

- Quan hệ **TheLoai**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDTheLoai	tinyint		PK	
TenTheLoai	nvarchar	15		

- Quan hệ **DinhDang**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
ID_DinhDang	tinyint		PK	
TenDinhDang	varchar	5		

- Quan hệ **NgonNgu**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
------------	--------------	---------	-----------	-----------

ID_NgonNgu	tinyint		PK	
NgonNgu	nvarchar	30		

- Quan hệ **DaoDien**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDDaoDien	int		PK	
TenDaoDien	nvarchar	20		

- Quan hệ **DienVien**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDDienVien	int		PK	
TenDienVien	nvarchar	20		

- Quan hệ **P\_DaoDien**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDPhim	int		FK	
IDDaoDien	int		FK	

- Quan hệ **P\_DienVien**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDPhim	int		FK	
IDDienVien	int		FK	

- Quan hệ **P\_TheLoai**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDPhim	int		FK	
IDTheLoai	tinyint		FK	

- Quan hệ **P\_DN**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
ID_PD_N	int		PK	
IDPhim	int		FK	

ID_DinhDang	tinyint		FK	
ID_NgonNgu	tinyint		FK	

- Quan hệ **LichChieuPhim**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
ID_LichChieu	int		PK	
ID_PDN	int		FK	
IDPhong	tinyint		FK	
ThoiGianChieu	smalldatetime			

- Quan hệ **Account\_KH**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDKhachHang	varchar		PK	
TenDangNhap	varchar			
MatKhai	varchar			

- Quan hệ **KhachHang**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDKhachHang	varchar	9	FK	
HoTen	nvarchar	30		
NgaySinh	date			
GioiTinh	nvarchar	3		

- Quan hệ **NhanVien**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDNhanVien	int		PK	
HoTen	nvarchar	30		
NgaySinh	date			
Que	nvarchar	20		
SoChungMinhThu	varchar	15		
GioiTinh	nvarchar	3		

- Quan hệ **ChucVu**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDChucVu	tinyint		PK	
ChucVu	nvarchar	20		

- Quan hệ **Account\_NV**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
IDNhanVien	int		PK	
TenDangNhap	varchar	20		
MatKhau	varchar	16		
IDChucVu	tinyint		FK	

- Quan hệ **ThoiGian**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
ID_TG	tinyint		PK	
TG_BatDau	time			
TG_KetThuc	time			

- Quan hệ **DonGia**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
ID_DinhDang	tinyint		PK	
Thu	tinyint			
ID_TG	tinyint		FK	
DonGia	real			

- Quan hệ **Booking\_POS**

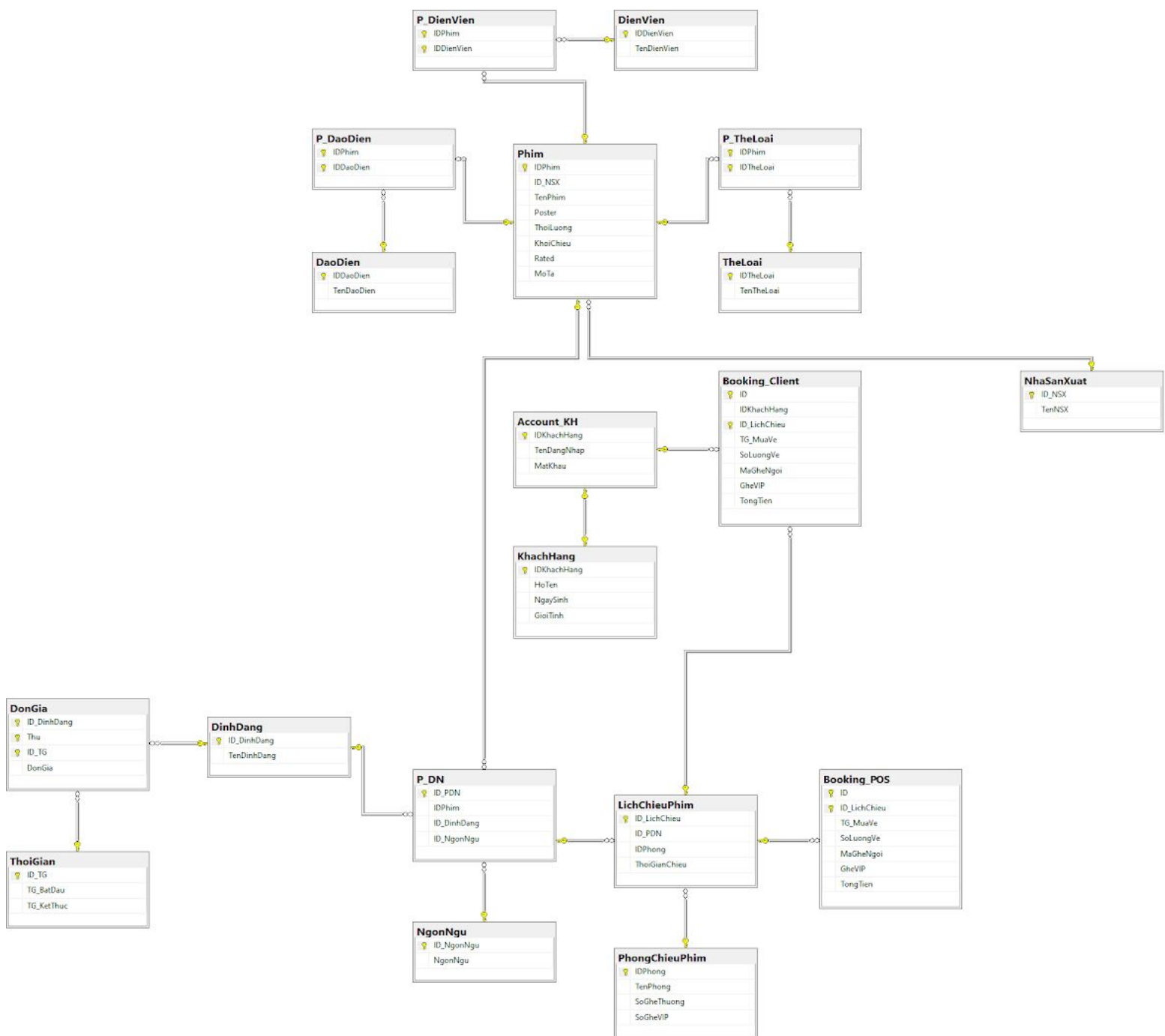
Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
ID	int		PK	
ID_LichChieu	int		FK	
TG_MuaVe	smalldatetime			
SoLuongVe	tinyint			

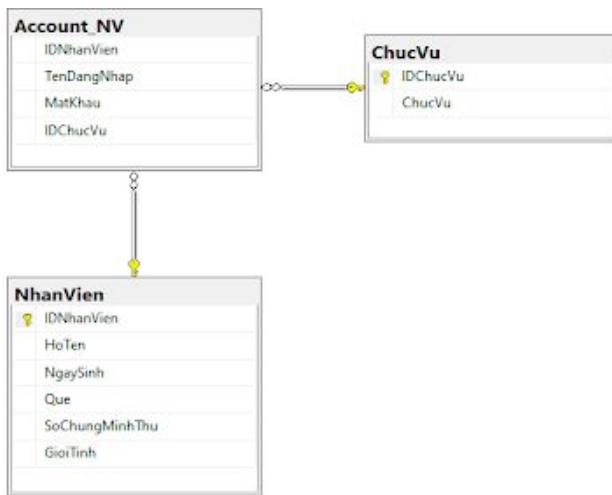
MaGheNgoi	varchar	VARCHAR(MAX)		
GheVIP	bit			
TongTien	real			

- Quan hệ **Booking\_Client**

Tên trường	Kiểu dữ liệu	Độ rộng	Ràng buộc	Chú thích
ID	varchar	6	PK	
IDKhachHang	varchar	9	FK	
ID_LichChieu	int		FK	
TG_MuaVe	smalldatetime			
SoLuongVe	tinyint			
MaGheNgoi	varchar			
GheVIP	bit			
TongTien	real			

### 3. Mô hình quan hệ ERD





## Chương 3: Cài đặt cơ sở dữ liệu

### 1. Tạo cơ sở dữ liệu

```

CREATE DATABASE QLRapChieuPhim
On Primary
(
    Name = QLRapChieuPhim_Data,
    Filename = 'D:\Program Files\Microsoft SQL
Server\MSSQL14.MSSQLSERVER\MSSQL\Data\QLRapChieuPhim_data.mdf',
    Size = 10 MB,
    MaxSize = 40 MB,
    FileGrowth = 1 MB
)
Log On
(
    Name = QLRapChieuPhim_Log,
    Filename = 'D:\Program Files\Microsoft SQL
Server\MSSQL14.MSSQLSERVER\MSSQL\Data\QLRapChieuPhim_log.ldf',
    Size = 10 MB,
    MaxSize = 20 MB,
    FileGrowth = 1 MB
)
Use QLRapChieuPhim
  
```



## 2. Tạo các bảng và ràng buộc dữ liệu trên bảng

```
CREATE TABLE PhongChieuPhim
```

```
(  
    IDPhongChieuPhim tinyint NOT NULL PRIMARY KEY,  
    TenPhong varchar(5),  
    SoGheThuong tinyint,  
    SoGheVIP tinyint,  
)
```

```
CREATE TABLE NhaSanXuat
```

```
(  
    ID_NSX int NOT NULL PRIMARY KEY Identity(1,1),  
    TenNSX nvarchar(20)  
)
```

```
CREATE TABLE TheLoai
```

```
(  
    IDTheLoai tinyint NOT NULL PRIMARY KEY Identity(1,1),  
    TenTheLoai nvarchar(15)  
)
```

```
CREATE TABLE DinhDang -- Định dạng phim: 2D, 3D...
```

```
(  
    ID_DinhDang tinyint NOT NULL PRIMARY KEY Identity(1,1),  
    TenDinhDang varchar(5)  
)
```

```
CREATE TABLE NgonNgu -- Ngôn ngữ phim: phụ đề việt, lồng Tiếng Việt...
```

```
(  
    ID_NgonNgu tinyint NOT NULL PRIMARY KEY Identity(1,1),  
    NgonNgu nvarchar(30)  
)
```

```
CREATE TABLE DaoDien
```

```
(  
    IDDaoDien int NOT NULL PRIMARY KEY Identity(1,1),  
    TenDaoien nvarchar(20)  
)
```

```
CREATE TABLE DienVien
```

```
(  
    IDDienVien int NOT NULL PRIMARY KEY Identity(1,1),  
    TenDienVien nvarchar(20)  
)
```

```
CREATE TABLE Phim
```

```
(
```

```

IDPhim int NOT NULL PRIMARY KEY,
ID_NSX int,
TenPhim nvarchar(max),
Poster image,
ThoiLuong tinyint,
KhoiChieu date,
Rated char(3),
MoTa nvarchar(max),
FOREIGN KEY (ID_NSX) REFERENCES NhaSanXuat(ID_NSX),
)

```

Create Table P\_DaoDien

```

(
    IDPhim int NOT NULL,
    IDDaoDien int NOT NULL,
    Primary key (IDPhim, IDDaoDien),

    FOREIGN KEY (IDPhim) REFERENCES Phim(IDPhim),
    FOREIGN KEY (IDDaoDien) REFERENCES DaoDien(IDDaoDien)
)

```

Create Table P\_DienVien

```

(
    IDPhim int NOT NULL,
    IDDienVien int NOT NULL,
    Primary key (IDPhim, IDDienVien),

    FOREIGN KEY (IDPhim) REFERENCES Phim(IDPhim),
    FOREIGN KEY (IDDienVien) REFERENCES DienVien(IDDienVien)
)

```

Create Table P\_TheLoai

```

(
    IDPhim int NOT NULL,
    IDTheLoai tinyint NOT NULL,
    Primary key (IDPhim, IDTheLoai),

    FOREIGN KEY (IDPhim) REFERENCES Phim(IDPhim),
    FOREIGN KEY (IDTheLoai) REFERENCES TheLoai(IDTheLoai)
)

```

Create Table P\_DN

```

(
    ID_PDN int NOT NULL PRIMARY KEY Identity(1,1),
    IDPhim int NOT NULL,
    ID_DinhDang tinyint NOT NULL,
    ID_NgonNgu tinyint NOT NULL,

    FOREIGN KEY (IDPhim) REFERENCES Phim(IDPhim),
    FOREIGN KEY (ID_DinhDang) REFERENCES DinhDang(ID_DinhDang),
    FOREIGN KEY (ID_NgonNgu) REFERENCES NgonNgu(ID_NgonNgu)
)

```

)

CREATE TABLE LichChieuPhim

(

ID\_LichChieu int NOT NULL PRIMARY KEY Identity(1,1),  
ID\_PDN int NOT NULL,  
IDPhongChieuPhim tinyint,  
ThoiGianChieu smalldatetime,

FOREIGN KEY (IDPhongChieuphim) REFERENCES PhongChieuphim(IDPhongChieuphim),  
FOREIGN KEY (ID\_PDN) REFERENCES P\_DN(ID\_PDN)

)

CREATE TABLE Account\_KH

(

IDKhachHang varchar(9) NOT NULL PRIMARY KEY,  
TenDangNhap varchar(20),  
MatKhau varchar(10),

)

CREATE TABLE KhachHang

(

IDKhachHang varchar(9) NOT NULL PRIMARY KEY,  
HoTen nvarchar(30),  
NgaySinh date,  
GioiTinh nvarchar(3) CHECK (GioiTinh= N'Nam' or GioiTinh= N'Nữ'),

FOREIGN KEY (IDKhachHang) REFERENCES Account\_KH(IDKhachHang)

)

CREATE TABLE NhanVien

(

IDNhanVien int NOT NULL PRIMARY KEY Identity(1,1),  
HoTen nvarchar(30),  
NgaySinh date,  
Que nvarchar(20),  
SoChungMinhThu varchar(15),  
GioiTinh nvarchar(3) CHECK (GioiTinh = N'Nam' or GioiTinh = N'Nữ'),

)

CREATE TABLE ChucVu

(

IDChucVu tinyint NOT NULL PRIMARY KEY Identity(1,1),  
ChucVu nvarchar(20),

)

CREATE TABLE Account\_NV

(

IDNhanVien int NOT NULL PRIMARY KEY,  
TenDangNhap varchar(20),  
MatKhau varchar(10),

```

        IDChucVu tinyint,

        FOREIGN KEY (IDNhanVien) REFERENCES NhanVien(IDNhanVien),
        FOREIGN KEY (IDChucVu) REFERENCES ChucVu(IDChucVu)
    )

CREATE TABLE ThoiGian
(
    ID_TG tinyint NOT NULL Primary key,
    TG_BatDau time,
    TG_KetThuc time
)

CREATE TABLE DonGia
(
    ID_DinhDang tinyint NOT NULL,
    Thu tinyint CHECK (Thu > 0 and Thu < 8),
    ID_TG tinyint,
    DonGia real,
    MoTa nvarchar(max),
    Primary key (ID_DinhDang, Thu, ID_TG),

    Foreign key (ID_DinhDang) References DinhDang(ID_DinhDang),
    Foreign key (ID_TG) References ThoiGian(ID_TG)
)

CREATE TABLE Booking_POS
(
    ID int NOT NULL,
    ID_LichChieu int NOT NULL,
    TG_MuaVe smalldatetime,
    SoLuongVe tinyint,
    MaGheNgoi varchar(max),
    GheVIP bit,
    TongTien real, --ThanhTien
    Primary key (ID, ID_LichChieu),

    FOREIGN KEY (ID_LichChieu) REFERENCES LichChieuPhim(ID_LichChieu)
)

CREATE TABLE Booking_Client
(
    ID varchar(6) NOT NULL,
    IDKhachHang varchar(9),
    ID_LichChieu int NOT NULL,
    TG_MuaVe smalldatetime,
    SoLuongVe tinyint,
    MaGheNgoi varchar(max),
    GheVIP bit,
    TongTien real,

```

Primary key (ID, ID\_LichChieu),

FOREIGN KEY (ID\_LichChieu) REFERENCES LichChieuPhim(ID\_LichChieu),

FOREIGN KEY (IDKhachHang) REFERENCES Account\_KH(IDKhachHang)

)

### 3. Tạo các view

Create View v\_DienVien\_Phim

As

```
Select P_DienVien.IDPhim, STRING_AGG(DienVien.TenDienVien, '  
) As TenDienVien
```

```
From DienVien, P_DienVien
```

```
Where IDPhim in (
```

```
Select * from dbo.getIDPhimDangChieu()
```

```
UNION ALL Select * from
```

```
dbo.getIDPhimSapChieu()
```

```
-- UNION ALL chạy nhanh hơn lệnh UNION vì nó không kiểm  
tra các bản ghi trùng lặp
```

```
)
```

```
and DienVien.IDDienVien = P_DienVien.IDDienVien
```

```
Group by P_DienVien.IDPhim
```

GO

Create View v\_DaoDien\_Phim

As

```
Select pdv.IDPhim, STRING_AGG(TenDaoDien, ', ') TenDaoDien
```

```
From DaoDien dd, P_DaoDien pdv
```

```
Where IDPhim in (
```

```
Select * from dbo.getIDPhimDangChieu()
```

```
UNION ALL Select * from
```

```
dbo.getIDPhimSapChieu()
```

```
)
```

```
and dd.IDDaoDien = pdv.IDDaoDien
```

```
Group by pdv.IDPhim
```

GO

Create View v\_TheLoai\_Phim

As

```
Select ptl.IDPhim, STRING_AGG(TenTheLoai, ', ') TenTheLoai
```

```
From TheLoai tl, P_TheLoai ptl
```

```
Where IDPhim in (
```

```

        Select * from dbo.getIDPhimDangChieu()
    UNION ALL Select * from dbo.getIDPhimSapChieu() )
        and tl.IDTheLoai = ptl.IDTheLoai
        Group by ptl.IDPhim

GO

Create View v_DinhDangNN
As
    Select pdn.ID_PD_N, IDPhim,
           nn.ngonngu, dd.TenDinhDang
    From P_DN pdn, NgonNgu nn, DinhDang dd
    Where pdn.ID_NgonNgu = nn.ID_NgonNgu
    and pdn.ID_DinhDang = dd.ID_DinhDang

-- View lấy các dữ liệu liên quan của 1 bộ phim
Create View v_DuLieu_Phim
As
    Select p.IDPhim, TenPhim, TenNSX, Poster,
           TenDienVien as DienVien, TenDaoDien as DaoDien, TenTheLoai
as TheLoai, TenDinhDang as DinhDang, NgonNgu, KhoiChieu, Thoiluong,
Rated, MoTa
    From Phim p, NhaSanXuat nsx, v_DienVien_Phim pdv,
v_DaoDien_Phim pdd, v_TheLoai_Phim vtl, v_DinhDangNN vdn
    Where p.IDPhim = pdv.IDPhim
    and p.ID_NSX = nsx.ID_NSX
    and p.IDPhim = pdd.IDPhim
    and p.IDPhim = vtl.IDPhim
    and p.IDPhim = vdn.IDPhim

GO

-- View lấy lịch chiếu phim
Create View v_LichChieuPhim
As
    Select IDPhim, ID_LichChieu, lcp.ID_PD_N , TenPhong, ThoiGianChieu,
           TenDinhDang, NgonNgu

```

```
From LichChieuPhim lcp, PhongChieuPhim pcp,  
v_DinhDangNN vdd
```

```
Where ThoiGianChieu >= GETDATE()  
and lcp.IDPhong = pcp.IDPhong  
and lcp.ID_PDN = vdd.ID_PDN
```

```
GO
```

```
-- View lấy các dữ liệu liên quan của 1 khách hàng
```

```
Create View v_dl_KhachHang (IDKhachHang, HoTen, NgaySinh, GioiTinh)
```

```
As
```

```
select kh.IDKhachHang, HoTen, NgaySinh, GioiTinh  
From KhachHang kh, Account_KH ackh  
where kh.IDKhachHang = ackh.IDKhachHang
```

```
GO
```

```
-- View lấy các dữ liệu liên quan của 1 nhân viên
```

```
Create View v_dl_Nhanvien
```

```
As
```

```
Select nv.IDNhanVien, HoTen, NgaySinh, Que, SoChungMinhThu,  
GioiTinh, ChucVu
```

```
From NhanVien nv, Account_NV a, ChucVu cv  
Where nv.IDNhanVien = a.IDNhanVien  
and cv.IDChucVu = a.IDChucvu
```

```
GO
```

```
-- View tổng hợp đơn giá
```

```
Create View v_DonGia
```

```
As
```

```
Select TenDinhDang, Thu, TG_BatDau, TG_KetThuc, DonGia  
From ThoiGian g, DonGia d, DinhDang dd  
Where g.ID_TG = d.ID_TG  
and dd.ID_DinhDang = d.ID_DinhDang
```



#### 4. Tạo các Store Procedure

-- Store Procedure accountLogin kiểm tra dữ liệu tài khoản NHÂN VIÊN đăng nhập có tồn tại không

```
Create Proc ckLogin_NV @userName varchar(20), @passWord varchar(16)
```

```
As
```

```
Begin
```

```
    Select HoTen, a.IDNhanVien
```

```
        From Account_NV a, NhanVien n
```

```
        Where TenDangNhap = @userName and MatKhai =
```

```
@passWord
```

```
        and a.IDNhanVien = n.IDNhanVien
```

```
End
```

```
GO
```

-- Store Procedure accountLogin kiểm tra dữ liệu tài khoản KHÁCH HÀNG đăng nhập có tồn tại không

```
Create Proc ckLogin_KH @userName varchar(20), @passWord varchar(16)
```

```
As
```

```
Begin
```

```
    Select a.IDKhachHang
```

```
        From Account_KH a, KhachHang n
```

```
        Where TenDangNhap = @userName and MatKhai =
```

```
@passWord
```

```
        and a.IDKhachHang = n.IDKhachHang
```

```
End
```

```
GO
```

-- Store Procedure Thêm 1 khách hàng mới vào bảng Account\_KH khi khách hàng đăng ký tài khoản

```
Create Proc SignUp_KH @IDKhachHang varchar(9), @userName varchar(20),
```

```
@passWord varchar(16), @HoTen nvarchar(30), @NgaySinh date, @GioiTinh
```

```
nvarchar(3)
```

```
As
```

```
Begin
```

```

Insert into Account_KH
    Values (@IDKhachHang, @userName, @passWord)
Insert into KhachHang
    Values (@IDKhachHang, COALESCE(@HoTen, 'unknown'),
COALESCE(@NgaySinh, null), COALESCE(@GioiTinh, ''))
End

```

```

GO
-- Store Procedure lấy thông tin tổng hợp của 1 tài khoản nhân viên, trả về họ tên và id
của nhân viên đó
Create Proc Get_dsNV @IDChucVu tinyint
As
Begin
    Select HoTen, a.IDNhanVien
        From Account_NV a, NhanVien n
        Where TenDangNhap = @userName and MatKhai =
@passWord
        and a.IDNhanVien = n.IDNhanVien
End

```

```

-- Store Procedure trả về số ghế còn trống (thường, VIP) theo lịch chiếu của mỗi phim
-- THAM SỐ: ID_LichChieu
-- tổng hợp từ số vé trong cả 2 bảng ChiTietHD_POS & ChiTietHD_KH
GO
create proc st_ghetrong
As
begin
    declare @pos table (SoLuongVe int, MaGheNgoi nvarchar(max), GheVip bit)
    insert into @pos
    select sum (SoLuongVe) SoLuongVe, STRING_AGG (MaGheNgoi, ',' ) as
MaGhe, GheVIP
        from Booking_POS
        group by GheVIP

```

```

declare @client table (SoLuongVe int, MaGheNgoi nvarchar(max), GheVip bit)
insert into @client
select sum (SoLuongVe) SoLuongVe, STRING_AGG (MaGheNgoi, ', ') as
MaGhe, GheVIP
        from Booking_Client
        group by GheVIP

select sum (SoLuongVe) SoLuongVe,
        STRING_AGG (MaGheNgoi, ', ') as MaGhe, GheVIP
        from (select * from @client UNION ALL select * from @POS)
as t
        group by GheVIP

end

```

go

```

-- Store Proc nối dl DienVien với bảng P_DienVien
Create proc view_DienVien
as
        select P_DienVien.IDPhim, STRING_AGG( DienVien.TenDienVien, '
') as TenDienVien
        from DienVien, P_DienVien
        where DienVien.IDDienVien=P_DienVien.IDDienVien
        group by P_DienVien.IDPhim

```

GO

```

-- Store Proc nối dl DaoDien với bảng P_DaoDien
CREATE PROC view_DaoDien

```

```

as
    select pdv.IDPhim, STRING_AGG( TenDaoDien, ' ' ) TenDaoDien
    from DaoDien dd, P_DaoDien pdv
    where dd.IDDaoDien=pdv.IDDaoDien
    group by pdv.IDPhim

GO
-- Store Proc lấy dl phim đang / sắp chiếu từ view v_DuLieu_Phim
Create Proc Get_Movies @comingSoon bit = 0, @all bit = 0
As
if (@comingSoon = 0)
    Begin
        if (@all = 0)
            Select IDPhim, TenPhim, Poster, ThoiLuong, Rated From
v_DuLieu_Phim Where IDPhim in (Select * From dbo.getIDPhimDangChieu() )
        else
            Select * From v_DuLieu_Phim    Where IDPhim in (Select *
From dbo.getIDPhimDangChieu() )
        End
    else
        Begin
            if (@all = 0)
                Select IDPhim, TenPhim, Poster, ThoiLuong, Rated From
v_DuLieu_Phim Where IDPhim in (Select * From dbo.getIDPhimSapChieu() )
            else
                Select * From v_DuLieu_Phim    Where IDPhim in (Select *
From dbo.getIDPhimSapChieu() )
            End
        End

GO
-- Lấy thông tin chi tiết của 1 bộ phim theo id
Create Proc Get_MovieInfors @id int
As
    Select DaoDien, DienVien, TheLoai, KhoiChieu, NgonNgu, MoTa, DinhDang
From v_DuLieu_Phim Where IDPhim = @id

```

GO

-- Store Proc trả về lịch chiếu của 1 bộ phim từ view\_LichChieuPhim theo NGÀY + số ghế đã đặt

-- tham số đầu vào là IDPhim, Ngày mà khách hàng chọn

Create Proc Get\_LichChieu\_Ngay @IDPhim int, @Ngay date

As

```
Select v.ID_LichChieu, ThoiGianChieu as ThoiGian, TenDinhDang as
DinhDang, NgonNgu, TenPhong, SoLuongVe as SoGheTrong
      From v_LichChieuPhim v, (Select * From Get_GhePhim(@IDPhim)) g
      Where cast(ThoiGianChieu as date) = @Ngay and v.IDPhim =
@IDPhim and g.ID_LichChieu = v.ID_LichChieu
```

GO

-- Store Proc trả về lịch chiếu của 1 bộ phim từ view\_LichChieuPhim theo PHÒNG

-- tham số đầu vào là IDPhim, Phòng chiếu

Create Proc Get\_LichChieu\_Phong @IDPhim int = null, @TenPhong varchar(5) = "

As

```
if (@TenPhong = "")
      Select TenPhong, ThoiGianChieu, TenDinhDang, NgonNgu
      From v_LichChieuPhim
      Where IDPhim = @IDPhim
else if (@IDPhim is Null)
      Select TenPhim, ThoiGianChieu, TenDinhDang, NgonNgu
      From v_LichChieuPhim v, Phim p
      Where v.IDPhim = @IDPhim and TenPhong =
@TenPhong and v.IDPhim = p.IDPhim
else
      Select ID_LichChieu, ThoiGianChieu, TenDinhDang, NgonNgu
      From v_LichChieuPhim
      Where IDPhim = @IDPhim and TenPhong = @TenPhong
```

GO

```
-- Store Proc INSERT dl vào bảng liên quan đến 1 bộ phim với bảng có 2 cột dữ liệu
Create Proc isrt_MovieInf @TableName varchar(10), @IDPhim int, @data
varchar(20)
```

```
As
```

```
Begin
```

```
    Declare @d varchar(5), @q varchar(max)
```

```
    Declare c Cursor
```

```
        For Select value From string_split(@data, ',')
```

```
    Open c
```

```
    Fetch NEXT From c into @d
```

```
    While (@@FETCH_STATUS = 0)
```

```
        Begin
```

```
            Set @q = 'Insert into ' + @TableName +
```

```
                ' Values(' + @IDPhim + ', ' + @d + ')
```

```
            Exec(@q)
```

```
            Fetch NEXT From c into @d
```

```
        End
```

```
    Close c
```

```
    DeAllocate c
```

```
End
```

```
GO
```

```
-- INSERT Các thông tin liên quan của 1 bộ phim với bảng có 3 cột
```

```
Create Proc isrt_MovieInfs @TableName varchar(10), @ColNames varchar(20),
```

```
    @IDPhim int, @data1 varchar(20), @data2 varchar(20)
```

```
As
```

```
Begin
```

```
    Declare @d1 varchar(5), @d2 varchar(5), @q varchar(max)
```

```
    Declare c1 Cursor
```

```
        For Select value From string_split(@data1, ',')
```

```
    Declare c2 Cursor
```

```
        For Select value From string_split(@data2, ',')
```

```
    Open c1
```

```
    Open c2
```

```
    Fetch NEXT From c1 into @d1
```

```

Fetch NEXT From c1 into @d1
While (@@FETCH_STATUS = 0)
    Begin
        Set @q = 'Insert into ' + @TableName + '(' + @ColNames + ')' +
            ' Values(' + @IDPhim + ', ' + @d1 + ', ' + @d2 + ')'
        Exec(@q)
        Fetch NEXT From c1 into @d1
        Fetch NEXT From c1 into @d1
    End
Close c1
DeAllocate c1
Close c2
DeAllocate c2
End

```

```

GO
-- Store Proc INSERT 1 bộ phim
Create Proc isrt_Movie @IDPhim int, @TenPhim nvarchar(max), @Poster image =
null,
    @ThoiLuong tinyint, @KhoiChieu date, @Rated char(3), @MoTa
nvarchar(max),
    @ID_NSX int = null, @IDs_DienVien varchar(20), @IDs_DaoDien
varchar(20),
    @IDs_NgonNgu varchar(20), @IDs_DinhDang varchar(20), @IDs_TheLoai
varchar(20)
As
Begin
    Set NoCount ON;

    Insert into Phim
        Values (@IDPhim, @ID_NSX, @TenPhim, @Poster, @ThoiLuong,
@KhoiChieu, @Rated, @MoTa)

    Exec isrt_MovieInf 'P_DaoDien', @IDPhim, @IDs_DaoDien
    Exec isrt_MovieInf 'P_DienVien', @IDPhim, @IDs_DienVien

```

```

        Exec isrt_MovieInf 'P_TheLoai', @IDPhim, @IDs_TheLoai
        Exec isrt_MovieInfs 'P_DN', 'IDPhim, ID_DinhDang, ID_NgonNgu',
        @IDPhim, @IDs_DinhDang, @IDs_NgonNgu
    End

GO
-- DELETE 1 bộ phim
Create Proc del_Movie @IDPhim int
As
Begin
    Declare @inserted Table (ID_PDN int)

    Delete From P_DaoDien Where IDPhim = @IDPhim
    Delete From P_DienVien Where IDPhim = @IDPhim
    Delete From P_TheLoai Where IDPhim = @IDPhim
    Delete From P_DN OUTPUT deleted.ID_PDN into @inserted Where IDPhim
= @IDPhim
    Delete From Phim Where IDPhim = @IDPhim
    Delete From LichChieuPhim Where ID_PDN in (Select * From @inserted)
End

GO
-- Store Procedure Cập nhật thông tin của 1 bộ phim
Create Proc update_Movie @IDPhim int, @IDs_DienVien varchar(20) = null,
@IDs_DaoDien varchar(20) = null,
    @IDs_NgonNgu varchar(20) = null, @IDs_DinhDang varchar(20) = null,
@IDs_TheLoai varchar(20) = null
As
Begin
    Set NoCount ON;
    if (@IDs_DaoDien != null)
        Begin
            Delete From P_DaoDien Where IDPhim = @IDPhim
            Exec isrt_MovieInf 'P_DaoDien', @IDPhim, @IDs_DaoDien
        End
    if (@IDs_DienVien != null)
        Begin
            Delete From P_DienVien Where IDPhim = @IDPhim

```



```

        Exec isrt_MovieInf 'P_DienVien', @IDPhim, @IDs_DienVien
    End
    if (@IDs_TheLoai != null)
        Begin
            Delete From P_TheLoai Where IDPhim = @IDPhim
            Exec isrt_MovieInf 'P_TheLoai', @IDPhim, @IDs_TheLoai
        End
    if (@IDs_NgonNgu != null)
        Begin
            Delete From P_DN Where IDPhim = @IDPhim
            Exec isrt_MovieInfs 'P_DN', 'IDPhim, ID_DinhDang,
ID_NgonNgu', @IDPhim, @IDs_DinhDang, @IDs_NgonNgu
        End
    End
End

```

```

GO
-- Trả về Đơn giá của phim lựa chọn
Create Proc Get_DonGia @ID_LichChieu int, @TenDinhDang varchar(5)
As
    Select DonGia
        From v_LichChieuPhim l, v_DonGia d
        Where ID_LichChieu = @ID_LichChieu
        and d.TenDinhDang = @TenDinhDang
        and DatePart(WEEKDAY, ThoiGianChieu) = Thu
        and cast(ThoiGianChieu as Time) between TG_BatDau and
TG_KetThuc

```

## 5. Tạo các Function

```
-- Nối 2 bảng DienVien và P_DienVien
Create Function DienVien_Phim()
    RETURNS @tb Table (IDPhim int, TenDienVien nvarchar(20))
As
Begin
    Insert Into @tb
        Select P_DienVien.IDPhim, STRING_AGG(DienVien.TenDienVien, ', ') As
        TenDienVien
            From DienVien, P_DienVien
            Where DienVien.IDDienVien = P_DienVien.IDDienVien
            Group by P_DienVien.IDPhim

    Return
End

GO

-- Nối 2 bảng DaoDien và P_DaoDien
Create Function DaoDien_Phim()
    RETURNS Table
As
    Return Select pdv.IDPhim, STRING_AGG(TenDaoDien, ', ') TenDaoDien
        From DaoDien dd, P_DaoDien pdv
        Where dd.IDDaoDien = pdv.IDDaoDien
        Group by pdv.IDPhim

Go

-- Lấy id của những bộ phim đang chiếu
Create Function getIDPhimDangChieu()
    RETURNS @bien TABLE (IDPhim int)
As
    Begin
        Insert into @bien
            Select IDPhim
            From LichChieuPhim lcp, P_DN pdn
            Where ThoiGianChieu >= GETDATE()
            and lcp.ID_PDN = pdn.ID_PDN

        Return
    End

GO
```

```

-- Lấy id của những bộ phim sắp chiếu
Create Function getIDPhimSapChieu()
    RETURNS @bien TABLE (IDPhim int)
As
    Begin
        Insert into @bien
            Select IDPhim
            From Phim
            Where KhoiChieu >= Convert(date, GETDATE())
        Return
    End

GO
-- Lấy danh sách các ghế đã đặt
Create Function Get_GhePhim(@IDPhim int)
    RETURNS @t TABLE (ID_LichChieu int, SoLuongVe int)
As
    Begin
        declare @id table (ID_LichChieu int, SoGhe int)
        Insert into @id
            Select ID_LichChieu, Sum(SoGheThuong + SoGheVIP) as SoGhe
            From v_LichChieuPhim v, PhongChieuPhim p
            Where IDPhim = @IDPhim and v.TenPhong = p.TenPhong
            Group by ID_LichChieu

        declare @pc table (ID_LichChieu int, SoLuongVe int, SoLuongVe2 int)
        Insert into @pc
            select i.ID_LichChieu, sum(p.SoLuongVe) SoLuongVe, sum(c.SoLuongVe)
            SoLuongVe2
            from Booking_POS p, @id i, Booking_Client c
            Where p.ID_LichChieu in (Select ID_LichChieu From @id) and
            c.ID_LichChieu in (Select ID_LichChieu From @id)
            Group by i.ID_LichChieu
        UNION ALL Select ID_LichChieu, 0, 0 From @id

        declare @total table (ID_LichChieu int, SoLuongVe int)
        Insert into @total
            select p.ID_LichChieu, SoLuongVe + SoLuongVe2 as SoLuongVe
            from @pc p

        Insert into @t
            select i.ID_LichChieu, (Select SoGhe from @id) - Sum(SoLuongVe) as
            SoLuongVe
            from @total t, @id i
            Where t.ID_LichChieu = i.ID_LichChieu

```

```

                                group by i.ID_LichChieu
        return
End

GO
-- Lấy mã ghế
Create Function Get_MaGhe(@ID_LichChieu int)
    RETURNS VARCHAR(max)
As
    Begin
        declare @pos varchar(max)
            select @pos = STRING_AGG (MaGheNgoi,',')
                from Booking_POS
                Where ID_LichChieu = @ID_LichChieu

        declare @client varchar(max)
            select @client = STRING_AGG (MaGheNgoi,',')
                from Booking_Client
                Where ID_LichChieu = @ID_LichChieu
        Declare @result varchar(max) = Concat(@pos, ' , ', @client)
        return @result
    End

```