

ĐẠI HỌC BÁCH KHOA HÀ NỘI  
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

\*\*\*\*\* 📄 \*\*\*\*\*



# **BÁO CÁO BÀI TẬP LỚN CUỐI KÌ HỌC PHẦN**

**Thực hành cơ sở dữ liệu**

Chủ Đề: *Quản lí rạp chiếu phim*

Nhóm sinh viên thực hiện: Nhóm C12

Mã học phần: IT3290

Mã lớp: 139353

STT	Họ và tên	Mã số sinh viên
1	Nguyễn Đức Minh	20205101
2	Nguyễn Tài Khoa	20215066
3	Trần Khánh Duy	20215016

## I. Phân tích hệ thống quản lí rạp chiếu phim

### 1. Người sử dụng

- Khách hàng
- Nhân viên
- Quản lý

Cần xây dựng cơ sở dữ liệu đáp ứng nhu cầu của tất cả mọi người sử dụng hệ thống

### 2. Yêu cầu nghiệp vụ:

Thực hiện các yêu cầu cơ bản của rạp chiếu phim như

- Bán vé
- Nhập phim
- Sắp xếp lịch phim
- Thống kê doanh thu
- Đánh giá phim
- Tra cứu phim
- Tính giá vé

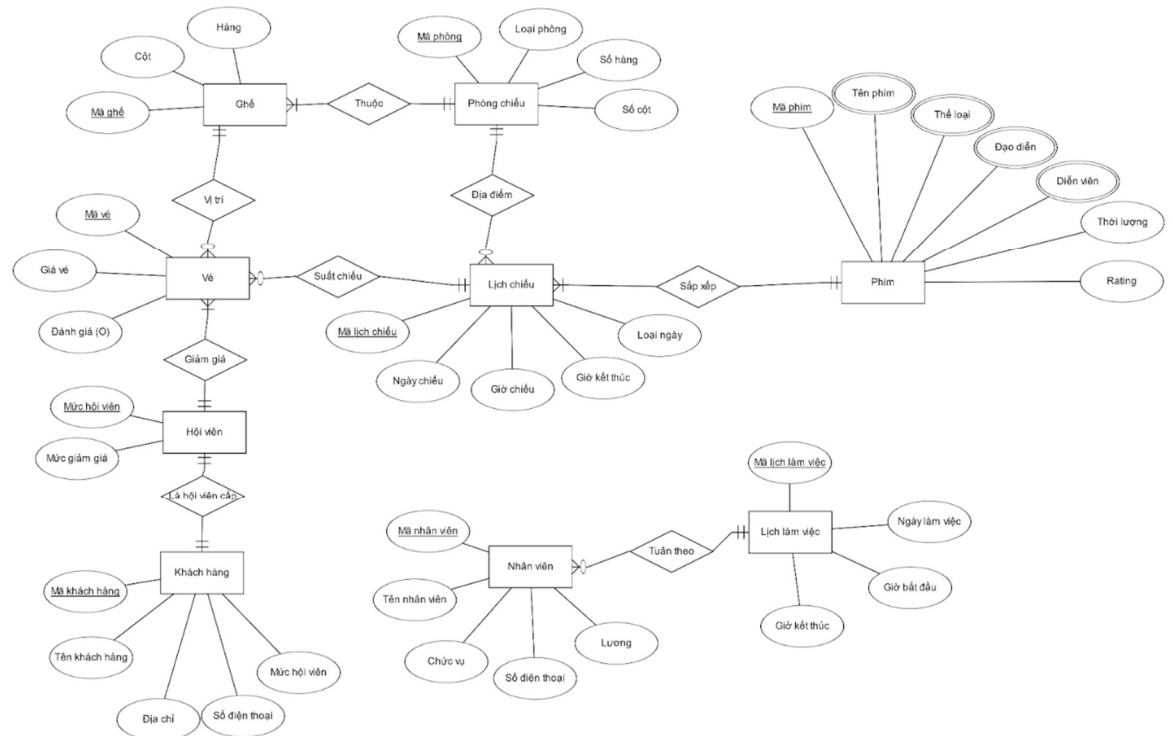
### 3. Yêu cầu về cơ sở dữ liệu

- Lưu trữ được các thông tin cần thiết với định dạng phù hợp như
  - + Tên phim, đạo diễn, diễn viên, thời lượng
  - + Tên khách hàng, số điện thoại, mức hội viên
  - + Tên nhân viên, lương, chức vụ giờ làm việc
- Đảm bảo tính chính xác và nhất quán của thông tin như:
  - + Không được bán vé cho phim không tồn tại hoặc không có lịch chiếu
  - + Không được bán nhiều hơn 1 vé cho cùng 1 vị trí ở cùng 1 suất chiếu
  - + Không được xếp 2 phim vào cùng 1 phòng chiếu trong cùng 1 khoảng thời điểm
  - + Nhân viên phải làm theo ca làm việc có trước và phải có lịch làm việc
- Thể hiện được các mối quan hệ và sự liên kết giữa dữ liệu như:

- + Vé cần có lịch chiếu và địa điểm cũng như số ghế để khách biết đến nơi
- + Nhân viên cần biết lịch làm việc như thế nào
- + Phim chiếu thì cần có phòng

## II. Thiết kế cơ sở dữ liệu

### 1. Sơ đồ thực thể liên kết



### 2. Phân tích sơ đồ và xây dựng cơ sở dữ liệu

- Có 9 thực thể
  - + Ghế
  - + Phòng chiếu
  - + Vé
  - + Lịch chiếu
  - + Phim
  - + Hội viên
  - + Khách hàng
  - + Nhân viên
  - + Lịch làm việc
- Từ các thực thể này ta sẽ tạo ra các bảng cùng tên với các thuộc tính tương ứng

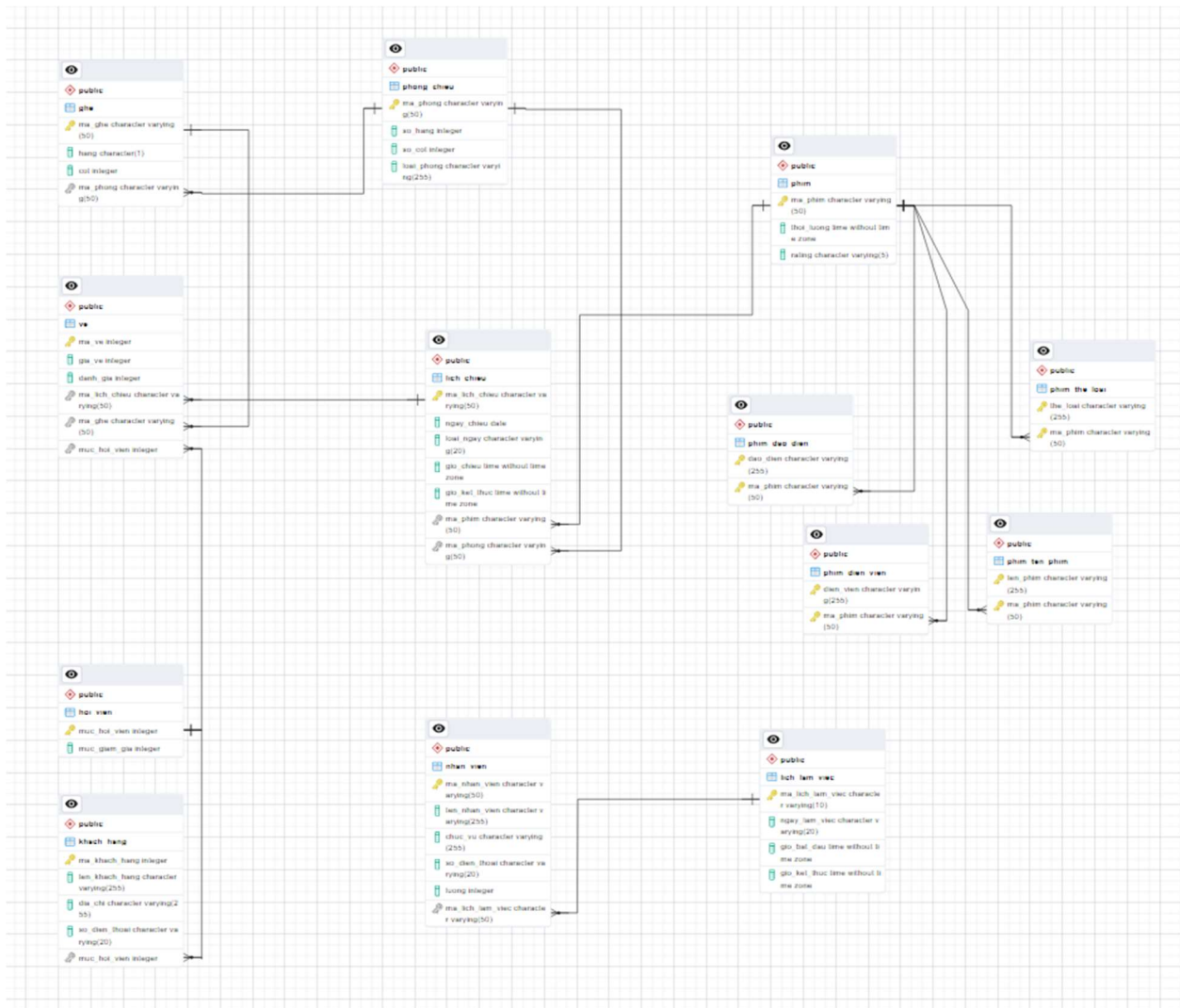
- Đối với các thuộc tính đa trị ta sẽ tạo bảng riêng rồi liên kết lại với thực thể gốc qua khóa ngoài

### 3. Bảng danh sách dữ liệu

Tên bảng	Tên cột	Kiểu dữ liệu	Độ rộng	Miêu tả
Hoi_vien	Muc_hoi_vien Muc_giam_gia	int int		Các mức hội viên và ưu đãi tương ứng
Phim	Ma_phim Thoi_luong Rating	Varchar Time Varchar	50 5	Các phim được chiếu trong rạp
Phong_chieu	Ma_phong So_hang So_cot Loai_phong	Varchar Int Int Varchar	50 255	Phòng chiếu và sức chứa của phòng Có mã tương ứng
Ghe	Ma_ghe Hang Cot Ma_phong	Varchar Char Int Varchar	50 1 50	Ghế và vị trí ghế tương ứng
Lich_lam_viec	Ma_lich_lam_viec Ngay_lam_viec Gio_bat_dau Gio_ket_thuc	Varchar Varchar Time Time	10 20	Thời gian biểu làm việc của nhân viên
Phim_Ten_phim	Ten_phim Ma_phim	Varchar Varchar	255 50	Tên phim ứng với mã phim tương ứng
Phim_The_loai	The_loai Ma_phim	Varchar Varchar	255 50	Thể loại phim ứng với mã phim tương ứng
Phim_Dao_dien	Dao_dien	Varchar	255	Đạo diễn của

	Ma_phim	Varchar	50	phim ứng với mã phim tương ứng
Phim_Dien_vien	Dao_dien Ma_phim	Varchar Varchar	255 50	Diễn viên của phim ứng với mã phim tương ứng
Lich_chieu	Ma_lich_chieu Ngay_chieu Loai_ngay Gio_chieu Gio_ket_thuc Ma_phim Ma_phong	Varchar Date Varchar Time Time Varchar Varchar	50  20   50 50	Lịch chiếu Của phim Có mã phim Tương ứng
Ve	Ma_ve Gia_ve Danh_gia Ma_lich_chieu Ma_ghe Muc_hoi_vien	Int Int Int Varchar Varchar Int	   50 50  	Vé, mã lịch Chiếu, giá vé, Đánh giá, Mã ghế của Phim tương ứng
Nhan_vien	Ma_nhan_vien Ten_nhan_vien Chuc_vu So_dien_thoai Luong Ma_lich_lam_viec	Varchar Varchar Varchar Varchar Int Varchar	50 255 255 20  50	Thông tin nhân viên
Khach_hang	Ma_khach_hang Ten_khach_hang Dia_chi So_dien_thoai Muc_hoi_vien	Int Varchar Varchar Varchar Int	 255 255 20  	Thông tin Khách hàng

#### 4. Sơ đồ quan hệ



## 5. Mã SQL cho cơ sở dữ liệu quản lý rạp chiếu phim

### a) Tạo bảng

```

-- Create sequence for Ma_lich_chieu
CREATE SEQUENCE lich_chieu_ma_lich_chieu_seq;

-- Create sequence for Ma_ve
CREATE SEQUENCE ve_ma_ve_seq;

-- Hoi_vien(Muc_hoi_vien, Muc_giam_gia)
CREATE TABLE Hoi_vien
(
    Muc_hoi_vien INT NOT NULL,
    Muc_giam_gia INT NOT NULL,
    PRIMARY KEY (Muc_hoi_vien)
);
  
```

```

-- Phim(Ma_phim, Thoi_luong, Rating)
CREATE TABLE Phim
(
    Ma_phim VARCHAR(50) NOT NULL,
    Thoi_luong TIME NOT NULL,
    Rating VARCHAR(5),
    PRIMARY KEY (Ma_phim)
);

-- Phong_chieu(Ma_phong, So_hang, So_cot, Loai_phong)
CREATE TABLE Phong_chieu
(
    Ma_phong VARCHAR(50) NOT NULL,
    So_hang INT NOT NULL CHECK (So_hang > 0 AND So_hang < 26),
    So_cot INT NOT NULL,
    Loai_phong VARCHAR(255) NOT NULL,
    PRIMARY KEY (Ma_phong)
);

-- Ghe(Ma_ghe, Hang, Cot, Ma_phong)
CREATE TABLE Ghe
(
    Ma_ghe VARCHAR(50) NOT NULL,
    Hang CHAR(1) NOT NULL CHECK (Hang >= 'A' AND Hang <= 'Z'),
    Cot INT NOT NULL,
    Ma_phong VARCHAR(50) NOT NULL,
    PRIMARY KEY (Ma_ghe),
    FOREIGN KEY (Ma_phong) REFERENCES Phong_chieu(Ma_phong)
);

-- Lich_lam_viec(Ma_lich_lam_viec, Ngay_lam_viec, Gio_bat_dau,
Gio_ket_thuc)
CREATE TABLE Lich_lam_viec
(
    Ma_lich_lam_viec VARCHAR(10) NOT NULL,
    Ngay_lam_viec VARCHAR(20) NOT NULL,
    Gio_bat_dau TIME NOT NULL,
    Gio_ket_thuc TIME NOT NULL,
    PRIMARY KEY (Ma_lich_lam_viec)
);

-- Phim_Ten_phim(Ten_phim, Ma_phim)
CREATE TABLE Phim_Ten_phim
(
    Ten_phim VARCHAR(255) NOT NULL,

```

```

Ma_phim VARCHAR(50) NOT NULL,
PRIMARY KEY (Ten_phim, Ma_phim),
FOREIGN KEY (Ma_phim) REFERENCES Phim(Ma_phim)
);

-- Phim_The_loai(The_loai, Ma_phim)
CREATE TABLE Phim_The_loai
(
    The_loai VARCHAR(255) NOT NULL,
    Ma_phim VARCHAR(50) NOT NULL,
    PRIMARY KEY (The_loai, Ma_phim),
    FOREIGN KEY (Ma_phim) REFERENCES Phim(Ma_phim)
);

-- Phim_Dao_dien(Dao_dien, Ma_phim)
CREATE TABLE Phim_Dao_dien
(
    Dao_dien VARCHAR(255) NOT NULL,
    Ma_phim VARCHAR(50) NOT NULL,
    PRIMARY KEY (Dao_dien, Ma_phim),
    FOREIGN KEY (Ma_phim) REFERENCES Phim(Ma_phim)
);

-- Phim_Dien_vien(Dien_vien, Ma_phim)
CREATE TABLE Phim_Dien_vien
(
    Dien_vien VARCHAR(255) NOT NULL,
    Ma_phim VARCHAR(50) NOT NULL,
    PRIMARY KEY (Dien_vien, Ma_phim),
    FOREIGN KEY (Ma_phim) REFERENCES Phim(Ma_phim)
);

-- Lich_chieu(Ma_lich_chieu, Ngay_chieu, Loai_ngay, Gio_chieu,
Gio_ket_thuc, Ma_phim, Ma_phong)
CREATE TABLE Lich_chieu
(
    Ma_lich_chieu VARCHAR(50) DEFAULT CONCAT('LC',
nextval('lich_chieu_ma_lich_chieu_seq')) NOT NULL,
    Ngay_chieu DATE NOT NULL,
    Loai_ngay VARCHAR(20) NOT NULL,
    Gio_chieu TIME NOT NULL,
    Gio_ket_thuc TIME NOT NULL,
    Ma_phim VARCHAR(50) NOT NULL,
    Ma_phong VARCHAR(50) NOT NULL,
    PRIMARY KEY (Ma_lich_chieu),

```



```

FOREIGN KEY (Ma_phim) REFERENCES Phim(Ma_phim),
FOREIGN KEY (Ma_phong) REFERENCES Phong_chieu(Ma_phong)
);

-- Ve(Ma_ve, Gia_ve, Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
CREATE TABLE Ve
(
    Ma_ve INT DEFAULT nextval('ve_ma_ve_seq') NOT NULL,
    Gia_ve INT NOT NULL,
    Danh_gia INT CHECK (Danh_gia BETWEEN 1 AND 5),
    Ma_lich_chieu VARCHAR(50) NOT NULL,
    Ma_ghe VARCHAR(50) NOT NULL,
    Muc_hoi_vien INT NOT NULL,
    PRIMARY KEY (Ma_ve),
    FOREIGN KEY (Ma_lich_chieu) REFERENCES Lich_chieu(Ma_lich_chieu),
    FOREIGN KEY (Ma_ghe) REFERENCES Ghe(Ma_ghe),
    FOREIGN KEY (Muc_hoi_vien) REFERENCES Hoi_vien(Muc_hoi_vien)
);

-- Khach_hang(Ma_khach_hang, Ten_khach_hang, Dia_chi, So_dien_thoai,
Muc_hoi_vien)
CREATE TABLE Khach_hang
(
    Ma_khach_hang INT NOT NULL,
    Ten_khach_hang VARCHAR(255) NOT NULL,
    Dia_chi VARCHAR(255) NOT NULL,
    So_dien_thoai VARCHAR(20) NOT NULL,
    Muc_hoi_vien INT NOT NULL,
    PRIMARY KEY (Ma_khach_hang),
    FOREIGN KEY (Muc_hoi_vien) REFERENCES Hoi_vien(Muc_hoi_vien)
);

-- Nhan_vien(Ma_nhan_vien, Ten_nhan_vien, Chuc_vu, So_dien_thoai,
Luong, Ma_lich_lam_viec)
CREATE TABLE Nhan_vien
(
    Ma_nhan_vien VARCHAR(50) NOT NULL,
    Ten_nhan_vien VARCHAR(255) NOT NULL,
    Chuc_vu VARCHAR(255) NOT NULL,
    So_dien_thoai VARCHAR(20) NOT NULL,
    Luong INT NOT NULL,
    Ma_lich_lam_viec VARCHAR(50) NOT NULL,
    PRIMARY KEY (Ma_nhan_vien),
    FOREIGN KEY (Ma_lich_lam_viec) REFERENCES
Lich_lam_viec(Ma_lich_lam_viec)

```

```
);
```

## b) Các trigger và functions giúp cho việc nhập thông tin

```
-- Trigger and function to automatically add seats of Phong_chieu to Ghe table
--Function
CREATE OR REPLACE FUNCTION add_seats_to_ghe()
RETURNS TRIGGER AS $$
BEGIN
    FOR r IN 1..NEW.So_hang LOOP
        FOR c IN 1..NEW.So_cot LOOP
            INSERT INTO Ghe (Ma_ghe, Hang, Cot, Ma_phong)
            VALUES (CONCAT(NEW.Ma_phong, '_', CHR(64 + r)::CHAR(1), c), CHR(64 +
r)::CHAR(1), c, NEW.Ma_phong);
        END LOOP;
    END LOOP;

    RAISE NOTICE 'Seats have been successfully added to the Ghe table.';

    RETURN NULL;
END;
$$ LANGUAGE plpgsql;
--Trigger
CREATE TRIGGER add_seats_trigger
AFTER INSERT ON Phong_chieu
FOR EACH ROW
EXECUTE FUNCTION add_seats_to_ghe();
--Trigger and function to automatically add gio_ket_thuc to Lich_chieu
--Function
CREATE OR REPLACE FUNCTION calculate_gio_ket_thuc()
RETURNS TRIGGER AS
$$
DECLARE
    phim_thoi_luong INTERVAL;
BEGIN
    SELECT Thoi_luong INTO phim_thoi_luong FROM Phim WHERE Ma_phim = NEW.Ma_phim;
    NEW.Gio_ket_thuc := NEW.Gio_chieu + INTERVAL '10 minutes' + phim_thoi_luong;
    RETURN NEW;
END;
$$
LANGUAGE plpgsql;
--Trigger
CREATE TRIGGER tg_calculate_gio_ket_thuc
BEFORE INSERT OR UPDATE ON Lich_chieu
FOR EACH ROW
EXECUTE FUNCTION calculate_gio_ket_thuc();
```

```

--Trigger and function to ensure Lich_chieu doesn't have conflicts
--Function
CREATE OR REPLACE FUNCTION check_duplicate_lich_chieu()
    RETURNS TRIGGER AS
$$
DECLARE
    count_rows INT;
BEGIN
    SELECT COUNT(*) INTO count_rows FROM Lich_chieu
    WHERE Ngay_chieu = NEW.Ngay_chieu
        AND Ma_phong = NEW.Ma_phong
        AND (NEW.Gio_chieu >= Gio_chieu AND NEW.Gio_chieu < Gio_ket_thuc );

    IF count_rows > 0 THEN
        RAISE EXCEPTION 'The time slot and screening room are already in use.';
    RETURN NULL;
    END IF;

    RETURN NEW;
END;
$$
LANGUAGE plpgsql;
--Trigger
CREATE TRIGGER tg_check_duplicate_lich_chieu
    BEFORE INSERT ON Lich_chieu
    FOR EACH ROW
    EXECUTE FUNCTION check_duplicate_lich_chieu();
--Trigger and Function to calculate ticket_price
--Function
CREATE OR REPLACE FUNCTION calculate_ticket_price()
    RETURNS TRIGGER AS
$$
DECLARE
    ticket_price INT;
    hoi_vien_muc_giam_gia INT;
    loai_phong VARCHAR(255);
    loai_ngay VARCHAR(20);
BEGIN
    SELECT Muc_giam_gia INTO hoi_vien_muc_giam_gia FROM Hoi_vien WHERE Muc_hoi_vien
= NEW.Muc_hoi_vien;

    IF hoi_vien_muc_giam_gia IS NULL THEN
        RAISE EXCEPTION 'Invalid Muc_hoi_vien: %', NEW.Muc_hoi_vien;
    END IF;

```

```

SELECT Lich_chieu.Loai_ngay, Phong_chieu.Loai_phong INTO loai_ngay, loai_phong
FROM Lich_chieu
JOIN Phong_chieu ON Lich_chieu.Ma_phong = Phong_chieu.Ma_phong
WHERE Lich_chieu.Ma_lich_chieu = NEW.Ma_lich_chieu;

IF loai_ngay IS NULL OR loai_phong IS NULL THEN
    RAISE EXCEPTION 'Invalid Ma_lich_chieu: %', NEW.Ma_lich_chieu;
END IF;

IF loai_ngay = 'Normal' THEN
    ticket_price := 40;
ELSIF loai_ngay = 'Special' THEN
    ticket_price := 60;
ELSE
    RAISE EXCEPTION 'Invalid Loai_ngay: %', loai_ngay;
END IF;

ticket_price := ticket_price - hoi_vien_muc_giam_gia;

IF loai_phong = '2D' THEN
    ticket_price := ticket_price;
ELSIF loai_phong = '3D' THEN
    ticket_price := ticket_price + 20;
ELSIF loai_phong = 'IMAX' THEN
    ticket_price := ticket_price + 25;
ELSE
    RAISE EXCEPTION 'Invalid Loai_phong: %', loai_phong;
END IF;

NEW.Gia_ve := ticket_price;
RETURN NEW;
END;
$$
LANGUAGE plpgsql;
--Trigger
CREATE TRIGGER tg_calculate_ticket_price
BEFORE INSERT ON Ve
FOR EACH ROW
EXECUTE FUNCTION calculate_ticket_price();
--Trigger and function to check if a seat availability
--Function
CREATE OR REPLACE FUNCTION check_seat_availability()
    RETURNS TRIGGER AS
$$
BEGIN

```

```

IF EXISTS (
    SELECT 1 FROM Ve
    WHERE Ma_lich_chieu = NEW.Ma_lich_chieu
        AND Ma_ghe = NEW.Ma_ghe
) THEN
    RAISE NOTICE 'The seat is already taken.';
    RETURN NULL;
END IF;

RETURN NEW;
END;
$$
LANGUAGE plpgsql;
--Trigger
CREATE TRIGGER tg_check_seat_availability
    BEFORE INSERT ON Ve
    FOR EACH ROW
    EXECUTE FUNCTION check_seat_availability();

```

#### b) Tạo roles và cấp quyền

--cấp quyền cho khách hàng

```

create ROLE Khach_hang with login password 'Khach_hang';
--Quyền xem
grant select on all tables in schema public to Khach_hang;
REVOKE select on table Lich_lam_viec,Nhan_vien from Khach_hang;

```

Quyền sửa

```

grant update on table Khach_hang to Khach_hang;

```

--cấp quyền cho nhân viên

```

create role Nhan_vien with login password 'Nhan_vien';
--Quyền xem
GRANT SELECT ON ALL TABLES IN SCHEMA public TO Nhan_Vien;
REVOKE SELECT ON TABLE Lich_lam_viec FROM Nhan_Vien;
REVOKE SELECT ON TABLE Nhan_vien FROM Nhan_Vien;
--Quyền xóa
GRANT DELETE ON TABLE Khach_hang to Nhan_vien;
--Quyền sửa
GRANT update on TABLE Ve,Khach_hang to Nhan_vien;

```

--cấp quyền cho quản lí

```
create role Quan_li with login password 'Quan_li';
GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO Quan_li;
```

--Data

```
--Data
INSERT INTO Phong_chieu (Ma_phong, So_hang, So_cot, Loai_phong)
VALUES ('PC001', 15, 20, '2D'),
('PC002', 20, 20, '3D'),
('PC003', 20, 20, 'IMAX'),
('PC004', 15, 15, '2D'),
('PC005', 15, 15, '2D')
;

-- Add Muc_hoi_vien values 0, 1, and 2 with corresponding Muc_giam_gia values to
Hoi_vien table
INSERT INTO Hoi_vien (Muc_hoi_vien, Muc_giam_gia)
VALUES (0, 0),
(1, 5),
(2, 10);

-- Add Phim (Movie) record
INSERT INTO Phim (Ma_phim, Thoi_luong, Rating)
VALUES ('P001', '01:40:00', 'T16');

-- Add Phim_Ten_phim (Movie_Title) records
INSERT INTO Phim_Ten_phim (Ten_phim, Ma_phim)
VALUES ('Nhiệm Vụ: Bất Khả Thi Nghiệp Báo Phần Một', 'P001'),
('Mission: Impossible - Dead Reckoning Part One', 'P001');

-- Add Phim_The_loai (Movie_Genre) records
INSERT INTO Phim_The_loai (The_loai, Ma_phim)
VALUES ('Hành Động', 'P001'),
('Action', 'P001'),
('Phiêu Lưu', 'P001'),
('Adventure', 'P001'),
('Gay Cấn', 'P001'),
('Thriller', 'P001');

-- Add Phim_Dao_dien (Movie_Director) record
INSERT INTO Phim_Dao_dien (Dao_dien, Ma_phim)
VALUES ('Christopher McQuarrie', 'P001');

-- Add Phim_Dien_vien (Movie_Actor) records
```

```

INSERT INTO Phim_Dien_vien (Dien_vien, Ma_phim)
VALUES ('Tom Cruise', 'P001'),
      ('Rebecca Ferguson', 'P001'),
      ('Ving Rhames', 'P001'),
      ('Simon Pegg', 'P001'),
      ('Hayley Atwell', 'P001'),
      ('Vanessa Kirby', 'P001'),
      ('Esai Morales', 'P001');

-- Add Phim (Movie) record
INSERT INTO Phim (Ma_phim, Thoi_luong, Rating)
VALUES ('P002', '01:59:00', 'T13');

-- Add Phim_Ten_phim (Movie_Title) records
INSERT INTO Phim_Ten_phim (Ten_phim, Ma_phim)
VALUES ('Cô Thành Trong Gương', 'P002'),
      ('Lonely Castle In The Mirror', 'P002');

-- Add Phim_Dao_dien (Movie_Director) record
INSERT INTO Phim_Dao_dien (Dao_dien, Ma_phim)
VALUES ('hara keiichi', 'P002');

-- Add Phim_Dien_vien (Movie_Actor) records
INSERT INTO Phim_Dien_vien (Dien_vien, Ma_phim)
VALUES ('ami touma', 'P002'),
      ('Takumi Kitamura', 'P002'),
      ('ashida mana', 'P002');

-- Add Phim_The_loai (Movie_Genre) records
INSERT INTO Phim_The_loai (The_loai, Ma_phim)
VALUES ('Animation', 'P002');

-- Add Phim (Movie) record
INSERT INTO Phim (Ma_phim, Thoi_luong, Rating)
VALUES ('P003', '02:00:00', 'T13');

-- Add Phim_Ten_phim (Movie_Title) records
INSERT INTO Phim_Ten_phim (Ten_phim, Ma_phim)
VALUES ('INDIANA JONES & VÒNG QUAY ĐỊNH MỆNH', 'P003'),
      ('Indiana Jones and the Dial of Destiny', 'P003');

-- Add Phim_Dao_dien (Movie_Director) record
INSERT INTO Phim_Dao_dien (Dao_dien, Ma_phim)
VALUES ('James Mangold', 'P003');

```

```

-- Add Phim_Dien_vien (Movie_Actor) records
INSERT INTO Phim_Dien_vien (Dien_vien, Ma_phim)
VALUES ('Boyd Holbrook', 'P003'),
      ('Thomas Kretschmann', 'P003'),
      ('Mads Mikkelsen', 'P003'),
      ('Harrison Ford', 'P003'),
      ('Phoebe Waller-Bridge', 'P003'),
      ('Shaunette Renée Wilson', 'P003');

-- Add Phim_The_loai (Movie_Genre) records
INSERT INTO Phim_The_loai (The_loai, Ma_phim)
VALUES ('Action', 'P003'),
      ('Adventure', 'P003');

-- Add Phim (Movie) record
INSERT INTO Phim (Ma_phim, Thoi_luong, Rating)
VALUES ('P004', '01:40:00', 'T16');

-- Add Phim_Ten_phim (Movie_Title) records
INSERT INTO Phim_Ten_phim (Ten_phim, Ma_phim)
VALUES ('Transformers: Quái Thú Trỗi Dậy', 'P004'),
      ('Transformers: Rise Of The Beasts', 'P004');

-- Add Phim_Dao_dien (Movie_Director) record
INSERT INTO Phim_Dao_dien (Dao_dien, Ma_phim)
VALUES ('steven caple jr.', 'P004');

-- Add Phim_Dien_vien (Movie_Actor) records
INSERT INTO Phim_Dien_vien (Dien_vien, Ma_phim)
VALUES ('Anthony Ramos', 'P004'),
      ('dominique fishback', 'P004'),
      ('tobe nwigwe', 'P004'),
      ('Peter Cullen', 'P004'),
      ('Ron Perlman', 'P004');

-- Add Phim_The_loai (Movie_Genre) records
INSERT INTO Phim_The_loai (The_loai, Ma_phim)
VALUES ('Action', 'P004');

-- Add Phim (Movie) record
INSERT INTO Phim (Ma_phim, Thoi_luong, Rating)
VALUES ('P005', '02:30:00', 'T13');

-- Add Phim_Ten_phim (Movie_Title) records
INSERT INTO Phim_Ten_phim (Ten_phim, Ma_phim)

```



```

VALUES ('Oppenheimer', 'P005');

-- Add Phim_Dao_dien (Movie_Director) record
INSERT INTO Phim_Dao_dien (Dao_dien, Ma_phim)
VALUES ('Christopher Nolan', 'P005');

-- Add Phim_Dien_vien (Movie_Actor) records
INSERT INTO Phim_Dien_vien (Dien_vien, Ma_phim)
VALUES ('Cillian Murphy', 'P005'),
       ('Emily Blunt', 'P005'),
       ('Robert Downey Jr.', 'P005'),
       ('Matt Damon', 'P005'),
       ('florence pugh', 'P005'),
       ('rami malek', 'P005'),
       ('benny safdie', 'P005');

-- Add Phim_The_loai (Movie_Genre) records
INSERT INTO Phim_The_loai (The_loai, Ma_phim)
VALUES ('Crime', 'P005'),
       ('Action', 'P005'),
       ('Thriller', 'P005'),
       ('Biography', 'P005');

-- Add Lich_chieu (Showtime) records
-- Film: Nhiệm Vụ: Bất Khả Thi Nghiệp Báo Phần Một (P001)
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '08:00:00', 'P001', 'PC001');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '10:00:00', 'P001', 'PC001');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '14:00:00', 'P001', 'PC003');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '18:00:00', 'P001', 'PC002');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '22:00:00', 'P001', 'PC001');

INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '08:00:00', 'P001', 'PC001');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '10:00:00', 'P001', 'PC001');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '14:00:00', 'P001', 'PC003');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '18:00:00', 'P001', 'PC002');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)

```

```

VALUES ('2023-07-10', 'Normal', '22:00:00', 'P001', 'PC001');

INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '08:00:00', 'P001', 'PC001');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '10:00:00', 'P001', 'PC001');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '14:00:00', 'P001', 'PC003');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '18:00:00', 'P001', 'PC002');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '22:00:00', 'P001', 'PC001');

-- Film: Cô Thành Trong Gương (P002)
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '08:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '11:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '14:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '18:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '22:00:00', 'P002', 'PC005');

INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '08:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '11:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '14:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '18:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '22:00:00', 'P002', 'PC005');

INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '08:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '11:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '14:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '18:00:00', 'P002', 'PC005');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '22:00:00', 'P002', 'PC005');

```

```

VALUES ('2023-07-11', 'Normal', '22:00:00', 'P002', 'PC005');

-- Film: INDIANA JONES & VÒNG QUAY ĐỊNH MỆNH (P003)
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '08:00:00', 'P003', 'PC002');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '11:00:00', 'P003', 'PC002');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '14:00:00', 'P003', 'PC004');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '18:00:00', 'P003', 'PC004');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '22:00:00', 'P003', 'PC004');

INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Special', '08:00:00', 'P003', 'PC002');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Special', '11:00:00', 'P003', 'PC002');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Special', '14:00:00', 'P003', 'PC004');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Special', '18:00:00', 'P003', 'PC004');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Special', '22:00:00', 'P003', 'PC004');

INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Special', '08:00:00', 'P003', 'PC002');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Special', '11:00:00', 'P003', 'PC002');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Special', '14:00:00', 'P003', 'PC004');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Special', '18:00:00', 'P003', 'PC004');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Special', '22:00:00', 'P003', 'PC004');

-- Film: Transformers: Quái Thú Trỗi Dậy (P004)
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '08:00:00', 'P004', 'PC003');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '11:00:00', 'P004', 'PC003');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '14:00:00', 'P004', 'PC001');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '18:00:00', 'P004', 'PC001');

```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '22:00:00', 'P004', 'PC002');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '08:00:00', 'P004', 'PC003');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '11:00:00', 'P004', 'PC003');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '14:00:00', 'P004', 'PC001');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '18:00:00', 'P004', 'PC001');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '22:00:00', 'P004', 'PC002');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '08:00:00', 'P004', 'PC003');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '11:00:00', 'P004', 'PC003');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '14:00:00', 'P004', 'PC001');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '18:00:00', 'P004', 'PC001');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '22:00:00', 'P004', 'PC002');
```

```
-- Film: Oppenheimer (P005)
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '08:00:00', 'P005', 'PC004');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '11:00:00', 'P005', 'PC004');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '14:00:00', 'P005', 'PC002');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '18:00:00', 'P005', 'PC003');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-09', 'Special', '22:00:00', 'P005', 'PC003');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '08:00:00', 'P005', 'PC004');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '11:00:00', 'P005', 'PC004');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '14:00:00', 'P005', 'PC002');
```

```
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
```

```

VALUES ('2023-07-10', 'Normal', '18:00:00', 'P005', 'PC003');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-10', 'Normal', '22:00:00', 'P005', 'PC003');

INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '08:00:00', 'P005', 'PC004');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '11:00:00', 'P005', 'PC004');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '14:00:00', 'P005', 'PC002');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '18:00:00', 'P005', 'PC003');
INSERT INTO Lich_chieu (Ngay_chieu, Loai_ngay, Gio_chieu, Ma_phim, Ma_phong)
VALUES ('2023-07-11', 'Normal', '22:00:00', 'P005', 'PC003');

--Thêm dữ liệu vào bảng "Lich_lam_viec":
INSERT INTO Lich_lam_viec (Ma_lich_lam_viec, Ngay_lam_viec, Gio_bat_dau,
Gio_ket_thuc)
VALUES ('LLV001', '2,3,4,5', '07:00:00', '14:00:00'),
('LLV002', '2,3,4,5', '14:00:00', '22:00:00'),
('LLV003', '6,7,CN', '7:00:00', '14:00:00'),
('LLV004', '6,7,CN', '14:00:00', '22:00:00'),
('LLV005', '2,3,4,5', '07:00:00', '22:00:00'),
('LLV006', '6,7,CN', '07:00:00', '22:00:00'),
('LLV007', '2,3,4,5,6,7,CN', '07:00:00', '16:00:00');

--Thêm dữ liệu vào bảng "Nhan_vien":
INSERT INTO Nhan_vien (Ma_nhan_vien, Ten_nhan_vien, Chuc_vu, So_dien_thoai,
Luong, Ma_lich_lam_viec)
VALUES ('NV001', 'Ngo Đức Khanh', 'Quan ly', '0123456789', 10000, 'LLV007'),
('NV002', 'Vu Đức Thịnh', 'Nhan vien ban ve', '0987654321', 5000, 'LLV002'),
('NV003', 'Le Chi Dung', 'Nhan viên ban ve', '0123987456', 6000, 'LLV003'),
('NV004', 'Do Dai Quang', 'Bao ve', '0123987456', 6000, 'LLV006'),
('NV005', 'Me Le Chi', 'Ban hang', '0823973256', 6000, 'LLV004'),
('NV006', 'Thong Dao Chi', 'Bao ve', '0943114821', 5000, 'LLV005');

--Thêm dữ liệu vào bảng "Khach_hang":
INSERT INTO Khach_hang
(Ma_khach_hang,Ten_khach_hang,Dia_chi,So_dien_thoai,Muc_hoi_vien)
VALUES ('0000','Pham Hoang Thanh','So 325 - Quan Hai Ba Trung - Ha
Noi','0954859485','0'),
('0001','Pham Thanh Giang','So 64 - Quan Hai Ba Trung - Ha
Noi','0954859485','0'),
('0002','Tran Nam Giang','So 23 - Quan 3 - Ho Chi Minh','0954859485','2'),

```

```
('0003','Nguyen Thanh Phong','So 5 - Quan 9 - Thu Duc - Ho Chi Minh','0954859485','1'),
('0004','Vu Kim Ngan','Xa Hoang Giang - Huyen Thanh My - Ha Noi','0954859485','0'),
('0005','Hoang Kieu Thuy','So 85 - Quan Hoan Kiem - Ha Noi','0954859485','2'),
('0006','Mac Van Khoa','So 32 - Quan My Duc - Ha Noi','0954859485','0'),
('0007','Tran Bao Quy','Xom 8 - Hoang Thanh - Nghe Tinh','0954859485','1');
```

```
INSERT INTO Khach_hang (Ma_khach_hang, Ten_khach_hang, Dia_chi, So_dien_thoai, Muc_hoi_vien)
```

```
VALUES
```

```
 ('0008', 'Nguyen Van A', '123 Nguyen Du, Quan 1, TP. Ho Chi Minh', '0901234567', 1),
 ('0009', 'Tran Thi B', '456 Le Loi, Quan 2, TP. Ho Chi Minh', '0912345678', 2),
 ('0010', 'Le Van C', '789 Nguyen Trai, Quan 3, TP. Ho Chi Minh', '0923456789', 1),
 ('0011', 'Pham Thi D', '321 Le Duan, Quan 4, TP. Ho Chi Minh', '0934567890', 0),
 ('0012', 'Vo Van E', '567 Tran Hung Dao, Quan 5, TP. Ho Chi Minh', '0945678901', 2),
 ('0013', 'Truong Van F', '789 Le Van Sy, Quan 6, TP. Ho Chi Minh', '0956789012', 1),
 ('0014', 'Hoang Thi G', '234 Pham Ngu Lao, Quan 1, TP. Ho Chi Minh', '0967890123', 2),
 ('0015', 'Bui Van H', '567 Nguyen Thi Minh Khai, Quan 3, TP. Ho Chi Minh', '0978901234', 0),
 ('0016', 'Ngo Thi I', '890 Tran Quoc Thao, Quan 10, TP. Ho Chi Minh', '0989012345', 1),
 ('0017', 'Vu Van J', '123 Le Van Viet, Quan 7, TP. Ho Chi Minh', '0990123456', 2),
 ('0018', 'Nguyen Thi K', '456 Phan Xich Long, Quan Phu Nhuan, TP. Ho Chi Minh', '0912345678', 0),
 ('0019', 'Tran Van L', '789 Le Loi, Quan 5, TP. Ho Chi Minh', '0923456789', 1),
 ('0020', 'Le Thi M', '321 Tran Hung Dao, Quan 4, TP. Ho Chi Minh', '0934567890', 2),
 ('0021', 'Pham Van N', '567 Nguyen Dinh Chieu, Quan 3, TP. Ho Chi Minh', '0945678901', 0),
 ('0022', 'Vo Thi O', '890 Le Thanh Ton, Quan 1, TP. Ho Chi Minh', '0956789012', 1),
 ('0023', 'Truong Van P', '234 Cao Thang, Quan 10, TP. Ho Chi Minh', '0967890123', 2),
 ('0024', 'Hoang Van Q', '567 Ly Thai To, Quan 11, TP. Ho Chi Minh', '0978901234', 0),
 ('0025', 'Bui Thi R', '890 Nguyen Binh Khiem, Quan Phu Nhuan, TP. Ho Chi Minh', '0989012345', 1),
```

```
 ('0026', 'Ngo Van S', '123 Tran Quoc Toan, Quan 7, TP. Ho Chi Minh',  
'0990123456', 2),  
 ('0027', 'Vu Thi T', '456 Pham Hong Thai, Quan 5, TP. Ho Chi Minh',  
'0912345678', 0),  
 ('0028', 'Nguyen Van U', '789 Nguyen Hue, Quan 1, TP. Ho Chi Minh',  
'0923456789', 1);
```

--Thêm dữ liệu vào bảng "Ve":

```
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (3, 'LC1', 'PC001_A1', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (3, 'LC1', 'PC001_A2', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (3, 'LC1', 'PC001_A3', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (3, 'LC1', 'PC001_A4', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (4, 'LC1', 'PC001_A5', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (5, 'LC1', 'PC001_A6', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (4, 'LC1', 'PC001_A7', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (3, 'LC1', 'PC001_A8', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (4, 'LC1', 'PC001_A9', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (2, 'LC1', 'PC001_A10', 0);
```

```
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (3, 'LC1', 'PC001_B1', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (3, 'LC1', 'PC001_B2', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (5, 'LC1', 'PC001_B3', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (2, 'LC1', 'PC001_B4', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (4, 'LC1', 'PC001_B5', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (5, 'LC1', 'PC001_B6', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)  
VALUES (1, 'LC1', 'PC001_B7', 0);  
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
```



```
VALUES (3, 'LC1', 'PC001_B8', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (4, 'LC1', 'PC001_B9', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (2, 'LC1', 'PC001_B10', 0);

INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (3, 'LC1', 'PC001_C1', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (3, 'LC1', 'PC001_C2', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (3, 'LC1', 'PC001_C3', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (2, 'LC1', 'PC001_C4', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (4, 'LC1', 'PC001_C5', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (5, 'LC1', 'PC001_C6', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (1, 'LC1', 'PC001_C7', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (3, 'LC1', 'PC001_C8', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (4, 'LC1', 'PC001_C9', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (3, 'LC1', 'PC001_C10', 0);

INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (3, 'LC2', 'PC001_A1', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (3, 'LC2', 'PC001_A2', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (3, 'LC2', 'PC001_A3', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (3, 'LC2', 'PC001_A4', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (4, 'LC2', 'PC001_A5', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (5, 'LC2', 'PC001_A6', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (4, 'LC2', 'PC001_A7', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (3, 'LC2', 'PC001_A8', 0);
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (4, 'LC2', 'PC001_A9', 0);
```



```
INSERT INTO Ve (Danh_gia, Ma_lich_chieu, Ma_ghe, Muc_hoi_vien)
VALUES (2, 'LC2', 'PC001_A10', 0);
```

### c) Một số câu lệnh truy vấn

```
--Nguyễn Tài Khoa
--1. Xem thông tin phim
CREATE OR REPLACE FUNCTION print_movie_info(movie_name VARCHAR(100))
RETURNS VOID AS $$
DECLARE
    movie_name_lower VARCHAR(100);
    ma_phim VARCHAR(50);
    dao_dien VARCHAR(50);
    dien_vien VARCHAR(200);
    thoi_luong TIME;
    the_loai VARCHAR(50);
BEGIN
    -- Convert movie name to lowercase
    movie_name_lower := LOWER(movie_name);

    -- Get Ma_phim
    SELECT pt.Ma_phim INTO ma_phim
    FROM Phim_Ten_phim pt
    WHERE LOWER(pt.Ten_phim) = movie_name_lower;

    -- Get director name
    SELECT pd.Dao_dien INTO dao_dien
    FROM Phim_Dao_dien pd
    WHERE pd.Ma_phim = ma_phim;

    -- Get actor names
    SELECT string_agg(pd.Dien_vien, ', ') INTO dien_vien
    FROM Phim_Dien_vien pd
    WHERE pd.Ma_phim = ma_phim;

    -- Get movie duration
    SELECT p.Thoi_luong INTO thoi_luong
    FROM Phim p
    WHERE p.Ma_phim = ma_phim;

    -- Get movie genres
    SELECT string_agg(ptl.The_loai, ', ') INTO the_loai
    FROM Phim_The_loai ptl
    WHERE ptl.Ma_phim = ma_phim;

    -- Raise notice to print the movie information
    RAISE NOTICE '%', movie_name;
```

```

        RAISE NOTICE 'Dao dien: %', dao_dien;
        RAISE NOTICE 'Dien vien: %', dien_vien;
        RAISE NOTICE 'Thoi luong: %', thoi_luong;
        RAISE NOTICE 'The loai: %', the_loai;

        RETURN;
END;
$$ LANGUAGE plpgsql;

--2.Xem đánh giá trung bình của 1 phim
SELECT AVG(Danh_gia) AS Diem_xep_loai_phim
FROM Ve
WHERE Ma_lich_chieu IN (
    SELECT Ma_lich_chieu
    FROM Lich_chieu
    WHERE Ma_phim = 'P003'
);

--3.Xem các phim chiếu trong 1 ngày
SELECT p.Ma_phim, p.Ten_phim
FROM Phim p
JOIN Lich_chieu lc ON p.Ma_phim = lc.Ma_phim
WHERE lc.Ngay_chieu = '2023-07-10';

--4.Xem các suất chiếu đang có của 1 phim
SELECT lc.Ma_lich_chieu, lc.Ngay_chieu, lc.Gio_chieu, lc.Gio_ket_thuc,
pc.Ma_phong
FROM Lich_chieu lc
JOIN Phim_Ten_phim pt ON lc.Ma_phim = pt.Ma_phim
JOIN Phong_chieu pc ON lc.Ma_phong = pc.Ma_phong
WHERE pt.Ten_phim = 'Oppenheimer'
ORDER BY lc.Ngay_chieu, lc.Gio_chieu;

--5.Tìm các nhân viên làm việc trong ngày thu 2
SELECT *
FROM Nhan_vien
WHERE Ma_lich_lam_viec IN (
    SELECT Ma_lich_lam_viec
    FROM Lich_lam_viec
    WHERE Ngay_lam_viec LIKE '%2%'
);

--6.rigger và function tự động add ghế khi thêm phòng
CREATE OR REPLACE FUNCTION add_seats_to_ghe()
RETURNS TRIGGER AS $$

```

```

BEGIN
    FOR r IN 1..NEW.So_hang LOOP
        FOR c IN 1..NEW.So_cot LOOP
            INSERT INTO Ghe (Ma_ghe, Hang, Cot, Ma_phong)
            VALUES (CONCAT(NEW.Ma_phong, '_', CHR(64 + r)::CHAR(1), c),
            CHR(64 + r)::CHAR(1), c, NEW.Ma_phong);
        END LOOP;
    END LOOP;

    RAISE NOTICE 'Seats have been successfully added to the Ghe table.';

    RETURN NULL;
END;
$$ LANGUAGE plpgsql;
--Trigger
CREATE TRIGGER add_seats_trigger
AFTER INSERT ON Phong_chieu
FOR EACH ROW
EXECUTE FUNCTION add_seats_to_ghe();

--7.Trigger va function tu dong add gio_ket_thuc vao Lich_chieu
--Function
CREATE OR REPLACE FUNCTION calculate_gio_ket_thuc()
    RETURNS TRIGGER AS
$$
DECLARE
    phim_thoi_luong INTERVAL;
BEGIN
    SELECT Thoi_luong INTO phim_thoi_luong FROM Phim WHERE Ma_phim =
NEW.Ma_phim;
    NEW.Gio_ket_thuc := NEW.Gio_chieu + INTERVAL '10 minutes' +
    phim_thoi_luong;
    RETURN NEW;
END;
$$
LANGUAGE plpgsql;
--Trigger
CREATE TRIGGER tg_calculate_gio_ket_thuc
    BEFORE INSERT OR UPDATE ON Lich_chieu
    FOR EACH ROW
    EXECUTE FUNCTION calculate_gio_ket_thuc();

--8.Trigger va function tu dongkiem tra lich chieu khong trung
--Funciton
CREATE OR REPLACE FUNCTION check_duplicate_lich_chieu()

```

```

    RETURNS TRIGGER AS
$$
DECLARE
    count_rows INT;
BEGIN
    SELECT COUNT(*) INTO count_rows FROM Lich_chieu
    WHERE Ngay_chieu = NEW.Ngay_chieu
        AND Ma_phong = NEW.Ma_phong
        AND (NEW.Gio_chieu >= Gio_chieu AND NEW.Gio_chieu < Gio_ket_thuc );

    IF count_rows > 0 THEN
        RAISE EXCEPTION 'The time slot and screening room are already in use.';
    RETURN NULL;
    END IF;

    RETURN NEW;
END;
$$
LANGUAGE plpgsql;
--Trigger
CREATE TRIGGER tg_check_duplicate_lich_chieu
    BEFORE INSERT ON Lich_chieu
    FOR EACH ROW
    EXECUTE FUNCTION check_duplicate_lich_chieu();

--9.Trigger va function tu dong tinh gia ve
--Function
CREATE OR REPLACE FUNCTION calculate_ticket_price()
    RETURNS TRIGGER AS
$$
DECLARE
    ticket_price INT;
    hoi_vien_muc_giam_gia INT;
    loai_phong VARCHAR(255);
    loai_ngay VARCHAR(20);
BEGIN
    SELECT Muc_giam_gia INTO hoi_vien_muc_giam_gia FROM Hoi_vien WHERE
Muc_hoi_vien = NEW.Muc_hoi_vien;

    IF hoi_vien_muc_giam_gia IS NULL THEN
        RAISE EXCEPTION 'Invalid Muc_hoi_vien: %', NEW.Muc_hoi_vien;
    END IF;

    SELECT Lich_chieu.Loai_ngay, Phong_chieu.Loai_phong INTO loai_ngay,
loai_phong

```

```

FROM Lich_chieu
JOIN Phong_chieu ON Lich_chieu.Ma_phong = Phong_chieu.Ma_phong
WHERE Lich_chieu.Ma_lich_chieu = NEW.Ma_lich_chieu;

IF loai_ngay IS NULL OR loai_phong IS NULL THEN
    RAISE EXCEPTION 'Invalid Ma_lich_chieu: %', NEW.Ma_lich_chieu;
END IF;

IF loai_ngay = 'Normal' THEN
    ticket_price := 40;
ELSIF loai_ngay = 'Special' THEN
    ticket_price := 60;
ELSE
    RAISE EXCEPTION 'Invalid Loai_ngay: %', loai_ngay;
END IF;

ticket_price := ticket_price - hoi_vien_muc_giam_gia;

IF loai_phong = '2D' THEN
    ticket_price := ticket_price;
ELSIF loai_phong = '3D' THEN
    ticket_price := ticket_price + 20;
ELSIF loai_phong = 'IMAX' THEN
    ticket_price := ticket_price + 25;
ELSE
    RAISE EXCEPTION 'Invalid Loai_phong: %', loai_phong;
END IF;

NEW.Gia_ve := ticket_price;
RETURN NEW;
END;
$$
LANGUAGE plpgsql;
--Trigger
CREATE TRIGGER tg_calculate_ticket_price
BEFORE INSERT ON Ve
FOR EACH ROW
EXECUTE FUNCTION calculate_ticket_price();

--10.Trigger va function tu dong kiem tra ghe co trung khong
--Function
CREATE OR REPLACE FUNCTION check_seat_availability()
    RETURNS TRIGGER AS
$$
BEGIN

```

```

IF EXISTS (
    SELECT 1 FROM Ve
    WHERE Ma_lich_chieu = NEW.Ma_lich_chieu
        AND Ma_ghe = NEW.Ma_ghe
) THEN
    RAISE NOTICE 'The seat is already taken.';
    RETURN NULL;
END IF;

RETURN NEW;
END;
$$
LANGUAGE plpgsql;
--Trigger
CREATE TRIGGER tg_check_seat_availability
    BEFORE INSERT ON Ve
    FOR EACH ROW
    EXECUTE FUNCTION check_seat_availability();

--Nguyễn Đức Minh
--11. Tính doanh thu 1 phim
CREATE OR REPLACE FUNCTION caculate_ticket(movie_name VARCHAR(100))
RETURN int
$$
DECLARE
    movie_name_lower VARCHAR(100);
    cal integer;
BEGIN
    select into cal SUM(Gia_ve) from ve where(select pt.Ten_phim from
Phim_Ten_phim = movie_name )
    RETURN cal;
END;
$$
language plpgsql;

--12. Tính tổng doanh thu rạp
CREATE OR REPLACE FUNCTION caculate_cinema()
RETURN int
$$
DECLARE
    tong integer;
    TABLE
BEGIN
    select into tong SUM(Gia_ve) from ve
    RETURN tong;

```

```

END;
$$
language plpgsql;

--13. Tìm phim có suất chiếu 3D
SELECT DISTINCT ON (P.Ma_phim)
       P.Ma_phim,
       PT.Ten_phim
FROM Phim P
JOIN Phim_Ten_phim PT ON P.Ma_phim = PT.Ma_phim
JOIN Lich_chieu LC ON P.Ma_phim = LC.Ma_phim
JOIN Phong_chieu PC ON LC.Ma_phong = PC.Ma_phong
WHERE PC.Loai_phong = '3D'
ORDER BY P.Ma_phim, PT.Ten_phim;

--14. Các phim mà phải trên 13 tuổi mới xem được
SELECT DISTINCT ON (P.Ma_phim)
       P.Ma_phim,
       PT.Ten_phim
FROM Phim P
JOIN Phim_Ten_phim PT ON P.Ma_phim = PT.Ma_phim
WHERE P.Rating IN ('T13', 'T16', 'T18');

--15. Top 5 phim có điểm rating cao nhất
SELECT P.Ma_phim, PT.Ten_phim, AVG(Ve.Danh_gia) AS Diem_xep_loai_phim
FROM Phim P
JOIN Phim_Ten_phim PT ON P.Ma_phim = PT.Ma_phim
JOIN Lich_chieu LC ON P.Ma_phim = LC.Ma_phim
JOIN Ve ON LC.Ma_lich_chieu = Ve.Ma_lich_chieu
GROUP BY P.Ma_phim, PT.Ten_phim
ORDER BY AVG(Ve.Danh_gia) DESC
LIMIT 5;

--16. Khách hàng có nhà ở Hà Nội
SELECT *
FROM Khách_hang
WHERE Dia_chi LIKE '%Hà Nội%';

--17. Tính Lương cần trả
SELECT SUM(Luong) AS TotalSalary
FROM Nhan_vien;

--18. Tính lợi nhuận
CREATE OR REPLACE FUNCTION calculate_cinema()
RETURNS integer AS

```

```

$$
DECLARE
    total_revenue integer;
    total_salary integer;
BEGIN
    SELECT SUM(Gia_ve) INTO total_revenue FROM ve;
    SELECT SUM(luong) INTO total_salary FROM Nhan_vien;
    RETURN total_revenue - total_salary;
END;
$$
LANGUAGE plpgsql;

select calculate_cinema() as Tong_doanh_thu;

--19. Tìm các suất chiếu đang có trong khoảng thời gian nào đó
SELECT Lich_chieu.Ngay_chieu, Lich_chieu.Gio_chieu, Phim_Ten_phim.Ten_phim
FROM Lich_chieu
JOIN Phim ON Lich_chieu.Ma_phim = Phim.Ma_phim
JOIN Phim_Ten_phim ON Phim.Ma_phim = Phim_Ten_phim.Ma_phim
WHERE Lich_chieu.Ngay_chieu = '2023-07-11'
    AND Lich_chieu.Gio_chieu >= '14:00:00';

--20. Tìm phim theo tên đạo diễn
SELECT DISTINCT ON (P.Ma_phim)
    P.Ma_phim,
    PT.Ten_phim
FROM Phim_Dao_dien PD
JOIN Phim_Ten_phim PT ON PD.Ma_phim = PT.Ma_phim
JOIN Phim P ON P.Ma_phim = PD.Ma_phim
WHERE LOWER(PD.Dao_dien) = LOWER('Christopher Nolan')
ORDER BY P.Ma_phim, PT.Ten_phim;

-- Trần Khánh Duy
--21. Tìm khách hàng ở mức hội viên 2
SELECT *
FROM Khach_hang
WHERE muc_hoi_vien = 2;

--22. Tìm người giữ chức vụ quản lý
SELECT *
FROM Nhan_vien
WHERE Chuc_vu = 'quan_ly';

--23. In ra các phim cùng đánh giá của chúng
SELECT DISTINCT ON (P.Ma_phim)

```



```

        P.Ma_phim,
        PT.Ten_phim,
        AVG(V.Danh_gia) AS Diem_xep_loai_phim
FROM Phim_Dao_dien PD
JOIN Phim_Ten_phim PT ON PD.Ma_phim = PT.Ma_phim
JOIN Phim P ON P.Ma_phim = PD.Ma_phim
JOIN Lich_chieu L ON L.Ma_phim = P.Ma_phim
JOIN Ve V ON V.Ma_lich_chieu = L.Ma_lich_chieu
GROUP BY P.Ma_phim, PT.Ten_phim
ORDER BY P.Ma_phim, PT.Ten_phim;

--24. Tính số vé bán được của phim theo các định dạng
SELECT Phong_chieu.Loai_phong, COUNT(*) AS ticket_count
FROM Ve
JOIN Lich_chieu ON Ve.Ma_lich_chieu = Lich_chieu.Ma_lich_chieu
JOIN Phim ON Lich_chieu.Ma_phim = Phim.Ma_phim
JOIN Phong_chieu ON Lich_chieu.Ma_phong = Phong_chieu.Ma_phong
WHERE Phim.Ma_phim = 'P001'
GROUP BY Phong_chieu.Loai_phong;

--25. Vé bán xét theo loại ngày
SELECT Lich_chieu.Loai_ngay, COUNT(*) AS ticket_count
FROM Ve
JOIN Lich_chieu ON Ve.Ma_lich_chieu = Lich_chieu.Ma_lich_chieu
JOIN Phim ON Lich_chieu.Ma_phim = Phim.Ma_phim
GROUP BY Lich_chieu.Loai_ngay;

--26. Tìm phim dựa trên thể loại
SELECT DISTINCT ON (P.Ma_phim)
        P.Ma_phim,
        PT.Ten_phim
FROM Phim_The_loai PL
JOIN Phim_Ten_phim PT ON PL.Ma_phim = PT.Ma_phim
JOIN Phim P ON P.Ma_phim = PT.Ma_phim
WHERE LOWER(PL.The_loai) = LOWER('Action')
ORDER BY P.Ma_phim, PT.Ten_phim;

--27. Xếp hạng phim dựa trên doanh thu
SELECT PT.Ten_phim, SUM(V.Gia_ve) AS doanh_thu
FROM Phim_Ten_phim PT
JOIN Lich_chieu LC ON PT.Ma_phim = LC.Ma_phim
JOIN Ve V ON LC.Ma_lich_chieu = V.Ma_lich_chieu
GROUP BY PT.Ten_phim
ORDER BY SUM(V.Gia_ve) DESC
LIMIT 5;

```

```
--28. Xếp hạng phòng dựa trên sức chứa
SELECT Ma_phong, So_hang * So_cot AS suc_chua
FROM Phong_chieu
ORDER BY suc_chua DESC;
```

```
--29. In số ghế còn trống
SELECT Ghe.Ma_ghe
FROM Ghe
LEFT JOIN (
    SELECT Ma_ghe
    FROM Ve
    WHERE Ma_lich_chieu = 'LC1'
) AS V ON Ghe.Ma_ghe = V.Ma_ghe
WHERE V.Ma_ghe IS NULL
ORDER BY Ghe.Ma_ghe;
```

```
--30. Vé bán xét theo mức hội viên
SELECT Muc_hoi_vien, COUNT(*) AS ticket_count
FROM Ve
GROUP BY Muc_hoi_vien;
```