-- Xóa các bảng nếu tồn tại

DROP TABLE IF EXISTS public.invoice\_promotion CASCADE;

DROP TABLE IF EXISTS public.ticket CASCADE;

DROP TABLE IF EXISTS public.schedule\_seat CASCADE;

DROP TABLE IF EXISTS public.showtime CASCADE;

DROP TABLE IF EXISTS public.movie\_promotion CASCADE;

DROP TABLE IF EXISTS public.movie\_version CASCADE;

DROP TABLE IF EXISTS public.movie\_type CASCADE;

DROP TABLE IF EXISTS public.movie\_review CASCADE;

DROP TABLE IF EXISTS public.invoice CASCADE;

DROP TABLE IF EXISTS public.promotion CASCADE;

DROP TABLE IF EXISTS public.version CASCADE;

DROP TABLE IF EXISTS public.type CASCADE;

DROP TABLE IF EXISTS public.seat CASCADE;

DROP TABLE IF EXISTS public.ticket\_type CASCADE;

DROP TABLE IF EXISTS public.cinema\_room CASCADE;

DROP TABLE IF EXISTS public.schedule CASCADE;

DROP TABLE IF EXISTS public.show\_dates CASCADE;

DROP TABLE IF EXISTS public.movie CASCADE;

DROP TABLE IF EXISTS public.member CASCADE;

DROP TABLE IF EXISTS public.employee CASCADE;

DROP TABLE IF EXISTS public.account CASCADE;

DROP TABLE IF EXISTS public.seat\_type CASCADE;

DROP TABLE IF EXISTS public.roles CASCADE;

DROP TABLE IF EXISTS public.movie\_age\_rating CASCADE;

BEGIN;

-- Bảng movie\_age\_rating: Lưu các mã phân loại độ tuổi

CREATE TABLE IF NOT EXISTS public.movie\_age\_rating (

rating\_code VARCHAR(10) PRIMARY KEY,

rating\_name VARCHAR(50) NOT NULL,

description TEXT

);

-- Bảng roles: Lưu các vai trò trong hệ thống

CREATE TABLE IF NOT EXISTS public.roles (

role\_id integer NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 2147483647 CACHE 1 ),

role\_name character varying(50) COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT roles\_pkey PRIMARY KEY (role\_id)

);

-- Bảng account: Lưu thông tin tài khoản người dùng

CREATE TABLE IF NOT EXISTS public.account (

account\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

username character varying(100) COLLATE pg\_catalog."default" NOT NULL,

password character varying(255) COLLATE pg\_catalog."default" NOT NULL,

full\_name character varying(150) COLLATE pg\_catalog."default" NOT NULL,

email character varying(100) COLLATE pg\_catalog."default" NOT NULL,

phone\_number character varying(20) COLLATE pg\_catalog."default",

address character varying(255) COLLATE pg\_catalog."default",

date\_of\_birth date,

gender character varying(1) COLLATE pg\_catalog."default",

identity\_card character varying(50) COLLATE pg\_catalog."default",

register\_date date DEFAULT CURRENT\_DATE,

status integer DEFAULT 1 CHECK (status IN (0, 1)),

role\_id integer NOT NULL,

CONSTRAINT account\_pkey PRIMARY KEY (account\_id),

CONSTRAINT account\_email\_key UNIQUE (email),

CONSTRAINT account\_username\_key UNIQUE (username),

CONSTRAINT fk\_account\_role FOREIGN KEY (role\_id)

REFERENCES public.roles (role\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng cinema\_room: Lưu thông tin các phòng chiếu

CREATE TABLE IF NOT EXISTS public.cinema\_room (

room\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

room\_name character varying(100) COLLATE pg\_catalog."default" NOT NULL UNIQUE,

seat\_quantity integer NOT NULL CHECK (seat\_quantity >= 0),

CONSTRAINT cinema\_room\_pkey PRIMARY KEY (room\_id)

);

-- Bảng seat\_type: Lưu các loại ghế

CREATE TABLE IF NOT EXISTS public.seat\_type (

seat\_type\_id integer NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 2147483647 CACHE 1 ),

type\_name character varying(50) COLLATE pg\_catalog."default" NOT NULL UNIQUE,

description character varying(255) COLLATE pg\_catalog."default",

CONSTRAINT seat\_type\_pkey PRIMARY KEY (seat\_type\_id)

);

-- Bảng seat: Lưu thông tin ghế trong phòng chiếu

CREATE TABLE IF NOT EXISTS public.seat (

seat\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

room\_id bigint NOT NULL,

seat\_row integer NOT NULL CHECK (seat\_row > 0),

seat\_column character varying(2) COLLATE pg\_catalog."default" NOT NULL,

seat\_type\_id integer NOT NULL,

is\_active boolean DEFAULT true,

CONSTRAINT seat\_pkey PRIMARY KEY (seat\_id),

CONSTRAINT uc\_seat\_room\_row\_column UNIQUE (room\_id, seat\_row, seat\_column),

CONSTRAINT fk\_seat\_room FOREIGN KEY (room\_id)

REFERENCES public.cinema\_room (room\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_seat\_type FOREIGN KEY (seat\_type\_id)

REFERENCES public.seat\_type (seat\_type\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng ticket\_type: Lưu các loại vé

CREATE TABLE IF NOT EXISTS public.ticket\_type (

ticket\_type\_id integer NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 2147483647 CACHE 1 ),

type\_name character varying(50) COLLATE pg\_catalog."default" NOT NULL UNIQUE,

description character varying(255) COLLATE pg\_catalog."default",

CONSTRAINT ticket\_type\_pkey PRIMARY KEY (ticket\_type\_id)

);

-- Bảng version: Lưu các phiên bản phim

CREATE TABLE IF NOT EXISTS public.version (

version\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

version\_name character varying(50) COLLATE pg\_catalog."default" NOT NULL UNIQUE,

description text COLLATE pg\_catalog."default",

CONSTRAINT version\_pkey PRIMARY KEY (version\_id)

);

-- Bảng movie: Lưu thông tin phim

CREATE TABLE IF NOT EXISTS public.movie (

movie\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

movie\_name character varying(255) COLLATE pg\_catalog."default" NOT NULL,

movie\_name\_en character varying(255) COLLATE pg\_catalog."default",

movie\_name\_vn character varying(255) COLLATE pg\_catalog."default",

director character varying(255) COLLATE pg\_catalog."default",

actor character varying(255) COLLATE pg\_catalog."default",

content text COLLATE pg\_catalog."default",

duration integer NOT NULL CHECK (duration > 0),

production\_company character varying(150) COLLATE pg\_catalog."default",

rating\_code character varying(10) COLLATE pg\_catalog."default",

from\_date date,

to\_date date,

large\_image\_url character varying(255) COLLATE pg\_catalog."default",

small\_image\_url character varying(255) COLLATE pg\_catalog."default",

trailer\_url character varying(255) COLLATE pg\_catalog."default",

CONSTRAINT movie\_pkey PRIMARY KEY (movie\_id),

CONSTRAINT chk\_date\_range CHECK (from\_date <= to\_date),

CONSTRAINT fk\_movie\_rating FOREIGN KEY (rating\_code)

REFERENCES public.movie\_age\_rating (rating\_code) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE RESTRICT

);

-- Bảng movie\_version: Liên kết phim và phiên bản

CREATE TABLE IF NOT EXISTS public.movie\_version (

movie\_id bigint NOT NULL,

version\_id bigint NOT NULL,

CONSTRAINT pk\_movie\_version PRIMARY KEY (movie\_id, version\_id),

CONSTRAINT fk\_movie\_version\_movie FOREIGN KEY (movie\_id)

REFERENCES public.movie (movie\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_movie\_version\_version FOREIGN KEY (version\_id)

REFERENCES public.version (version\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng movie\_review: Lưu đánh giá và nhận xét của người dùng

CREATE TABLE IF NOT EXISTS public.movie\_review (

review\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

movie\_id bigint NOT NULL,

account\_id bigint NOT NULL,

rating integer NOT NULL CHECK (rating >= 1 AND rating <= 5),

comment text COLLATE pg\_catalog."default",

review\_date timestamp without time zone DEFAULT CURRENT\_TIMESTAMP,

CONSTRAINT movie\_review\_pkey PRIMARY KEY (review\_id),

CONSTRAINT fk\_review\_movie FOREIGN KEY (movie\_id)

REFERENCES public.movie (movie\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_review\_account FOREIGN KEY (account\_id)

REFERENCES public.account (account\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng type: Lưu thể loại phim

CREATE TABLE IF NOT EXISTS public.type (

type\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

type\_name character varying(255) COLLATE pg\_catalog."default" NOT NULL UNIQUE,

CONSTRAINT type\_pkey PRIMARY KEY (type\_id)

);

-- Bảng movie\_type: Liên kết phim và thể loại

CREATE TABLE IF NOT EXISTS public.movie\_type (

movie\_id bigint NOT NULL,

type\_id bigint NOT NULL,

CONSTRAINT pk\_movie\_type PRIMARY KEY (movie\_id, type\_id),

CONSTRAINT fk\_movietype\_movie FOREIGN KEY (movie\_id)

REFERENCES public.movie (movie\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_movietype\_type FOREIGN KEY (type\_id)

REFERENCES public.type (type\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng promotion: Lưu thông tin khuyến mãi

CREATE TABLE IF NOT EXISTS public.promotion (

promotion\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

title character varying(150) COLLATE pg\_catalog."default" NOT NULL,

detail text COLLATE pg\_catalog."default",

discount\_level numeric(5,2) NOT NULL CHECK (discount\_level >= 0 AND discount\_level <= 100),

start\_time timestamp without time zone NOT NULL,

end\_time timestamp without time zone NOT NULL,

image\_url character varying(255) COLLATE pg\_catalog."default",

is\_active boolean DEFAULT true,

CONSTRAINT promotion\_pkey PRIMARY KEY (promotion\_id),

CONSTRAINT chk\_promo\_date CHECK (start\_time < end\_time)

);

-- Bảng movie\_promotion: Liên kết phim và khuyến mãi

CREATE TABLE IF NOT EXISTS public.movie\_promotion (

movie\_id bigint NOT NULL,

promotion\_id bigint NOT NULL,

CONSTRAINT pk\_movie\_promotion PRIMARY KEY (movie\_id, promotion\_id),

CONSTRAINT fk\_mp\_movie FOREIGN KEY (movie\_id)

REFERENCES public.movie (movie\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_mp\_promotion FOREIGN KEY (promotion\_id)

REFERENCES public.promotion (promotion\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng show\_dates: Lưu các ngày chiếu

CREATE TABLE IF NOT EXISTS public.show\_dates (

show\_date\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

show\_date date NOT NULL,

CONSTRAINT show\_dates\_pkey PRIMARY KEY (show\_date\_id),

CONSTRAINT unique\_show\_date UNIQUE (show\_date)

);

-- Bảng schedule: Lưu các khung giờ chiếu

CREATE TABLE IF NOT EXISTS public.schedule (

schedule\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

schedule\_time time without time zone NOT NULL,

CONSTRAINT schedule\_pkey PRIMARY KEY (schedule\_id)

);

-- Bảng showtime: Lưu suất chiếu

CREATE TABLE IF NOT EXISTS public.showtime (

showtime\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

movie\_id bigint NOT NULL,

show\_date\_id bigint NOT NULL,

schedule\_id bigint NOT NULL,

room\_id bigint NOT NULL,

version\_id bigint NOT NULL,

available\_seats integer CHECK (available\_seats >= 0),

room\_number character varying(50) COLLATE pg\_catalog."default",

CONSTRAINT showtime\_pkey PRIMARY KEY (showtime\_id),

CONSTRAINT uc\_showtime\_unique UNIQUE (show\_date\_id, schedule\_id, room\_id),

CONSTRAINT fk\_showtime\_movie FOREIGN KEY (movie\_id)

REFERENCES public.movie (movie\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_showtime\_date FOREIGN KEY (show\_date\_id)

REFERENCES public.show\_dates (show\_date\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_showtime\_schedule FOREIGN KEY (schedule\_id)

REFERENCES public.schedule (schedule\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_showtime\_room FOREIGN KEY (room\_id)

REFERENCES public.cinema\_room (room\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_showtime\_version FOREIGN KEY (version\_id)

REFERENCES public.version (version\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng schedule\_seat: Lưu thông tin ghế cho từng suất chiếu

CREATE TABLE IF NOT EXISTS public.schedule\_seat (

schedule\_seat\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

showtime\_id bigint NOT NULL,

seat\_id bigint NOT NULL,

seat\_price numeric(12,2) NOT NULL CHECK (seat\_price >= 0),

status integer DEFAULT 0 CHECK (status IN (0, 1, 2)),

CONSTRAINT schedule\_seat\_pkey PRIMARY KEY (schedule\_seat\_id),

CONSTRAINT uc\_schedule\_seat UNIQUE (showtime\_id, seat\_id),

CONSTRAINT fk\_ss\_showtime FOREIGN KEY (showtime\_id)

REFERENCES public.showtime (showtime\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_ss\_seat FOREIGN KEY (seat\_id)

REFERENCES public.seat (seat\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng invoice: Lưu thông tin hóa đơn

CREATE TABLE IF NOT EXISTS public.invoice (

invoice\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

account\_id bigint NOT NULL,

booking\_date timestamp without time zone DEFAULT CURRENT\_TIMESTAMP,

total\_amount numeric(12,2) NOT NULL CHECK (total\_amount >= 0),

payment\_method character varying(50) COLLATE pg\_catalog."default" CHECK (payment\_method IN ('credit\_card', 'online')),

use\_score integer DEFAULT 0 CHECK (use\_score >= 0),

add\_score integer DEFAULT 0 CHECK (add\_score >= 0),

status integer DEFAULT 1 CHECK (status IN (-1, 0, 1)),

movie\_name character varying(255) COLLATE pg\_catalog."default",

seat\_number character varying(50) COLLATE pg\_catalog."default",

CONSTRAINT invoice\_pkey PRIMARY KEY (invoice\_id),

CONSTRAINT fk\_invoice\_account FOREIGN KEY (account\_id)

REFERENCES public.account (account\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng invoice\_promotion: Liên kết hóa đơn và khuyến mãi

CREATE TABLE IF NOT EXISTS public.invoice\_promotion (

invoice\_id bigint NOT NULL,

promotion\_id bigint NOT NULL,

CONSTRAINT pk\_invoice\_promotion PRIMARY KEY (invoice\_id, promotion\_id),

CONSTRAINT fk\_ip\_invoice FOREIGN KEY (invoice\_id)

REFERENCES public.invoice (invoice\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_ip\_promotion FOREIGN KEY (promotion\_id)

REFERENCES public.promotion (promotion\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng ticket: Lưu thông tin vé

CREATE TABLE IF NOT EXISTS public.ticket (

ticket\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

invoice\_id bigint NOT NULL,

schedule\_seat\_id bigint NOT NULL,

ticket\_type\_id integer NOT NULL,

price numeric(12,2) NOT NULL CHECK (price >= 0),

CONSTRAINT ticket\_pkey PRIMARY KEY (ticket\_id),

CONSTRAINT uc\_ticket\_unique\_ss UNIQUE (schedule\_seat\_id),

CONSTRAINT fk\_ticket\_invoice FOREIGN KEY (invoice\_id)

REFERENCES public.invoice (invoice\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_ticket\_ss FOREIGN KEY (schedule\_seat\_id)

REFERENCES public.schedule\_seat (schedule\_seat\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE,

CONSTRAINT fk\_ticket\_type FOREIGN KEY (ticket\_type\_id)

REFERENCES public.ticket\_type (ticket\_type\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng member: Lưu thông tin thành viên

CREATE TABLE IF NOT EXISTS public.member (

member\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

account\_id bigint NOT NULL,

score integer DEFAULT 0 CHECK (score >= 0),

tier character varying(50) COLLATE pg\_catalog."default" DEFAULT 'Bạc' CHECK (tier IN ('Bạc', 'Vàng', 'Kim cương')),

CONSTRAINT member\_pkey PRIMARY KEY (member\_id),

CONSTRAINT member\_account\_id\_key UNIQUE (account\_id),

CONSTRAINT fk\_member\_account FOREIGN KEY (account\_id)

REFERENCES public.account (account\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Bảng employee: Lưu thông tin nhân viên

CREATE TABLE IF NOT EXISTS public.employee (

employee\_id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 9223372036854775807 CACHE 1 ),

account\_id bigint NOT NULL,

hire\_date date,

position character varying(100) COLLATE pg\_catalog."default",

salary numeric(12,2) CHECK (salary >= 0),

CONSTRAINT employee\_pkey PRIMARY KEY (employee\_id),

CONSTRAINT employee\_account\_id\_key UNIQUE (account\_id),

CONSTRAINT fk\_employee\_account FOREIGN KEY (account\_id)

REFERENCES public.account (account\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE CASCADE

);

-- Trigger để tự động cập nhật tier trong bảng member dựa trên score

CREATE OR REPLACE FUNCTION update\_member\_tier()

RETURNS TRIGGER AS $$

BEGIN

NEW.tier = CASE

WHEN NEW.score >= 5000 THEN 'Kim cương'

WHEN NEW.score >= 1000 THEN 'Vàng'

ELSE 'Bạc'

END;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER trigger\_update\_member\_tier

BEFORE INSERT OR UPDATE OF score

ON public.member

FOR EACH ROW

EXECUTE FUNCTION update\_member\_tier();

-- Trigger để tự động cập nhật available\_seats trong bảng showtime

CREATE OR REPLACE FUNCTION update\_available\_seats()

RETURNS TRIGGER AS $$

BEGIN

UPDATE public.showtime

SET available\_seats = (

SELECT COUNT(\*)

FROM public.schedule\_seat

WHERE showtime\_id = NEW.showtime\_id

AND status = 0

)

WHERE showtime\_id = NEW.showtime\_id;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER trigger\_update\_available\_seats

AFTER INSERT OR UPDATE OF status

ON public.schedule\_seat

FOR EACH ROW

EXECUTE FUNCTION update\_available\_seats();

-- Trigger để kiểm tra khoảng cách 30 phút giữa các suất chiếu trong cùng một phòng

CREATE OR REPLACE FUNCTION check\_showtime\_gap()

RETURNS TRIGGER AS $$

DECLARE

movie\_duration integer;

showtime\_end timestamp;

new\_start\_time timestamp;

conflicting\_showtime record;

BEGIN

SELECT duration INTO movie\_duration

FROM public.movie

WHERE movie\_id = NEW.movie\_id;

SELECT (sd.show\_date + s.schedule\_time) INTO new\_start\_time

FROM public.show\_dates sd

JOIN public.schedule s ON s.schedule\_id = NEW.schedule\_id

WHERE sd.show\_date\_id = NEW.show\_date\_id;

showtime\_end := new\_start\_time + (movie\_duration || ' minutes')::interval + interval '30 minutes';

FOR conflicting\_showtime IN (

SELECT sd.show\_date, s.schedule\_time, m.duration

FROM public.showtime st

JOIN public.show\_dates sd ON st.show\_date\_id = sd.show\_date\_id

JOIN public.schedule s ON st.schedule\_id = s.schedule\_id

JOIN public.movie m ON st.movie\_id = m.movie\_id

WHERE st.room\_id = NEW.room\_id

AND st.showtime\_id != NEW.showtime\_id

) LOOP

DECLARE

existing\_start timestamp;

existing\_end timestamp;

BEGIN

existing\_start := conflicting\_showtime.show\_date + conflicting\_showtime.schedule\_time;

existing\_end := existing\_start + (conflicting\_showtime.duration || ' minutes')::interval + interval '30 minutes';

IF new\_start\_time < existing\_end AND showtime\_end > existing\_start THEN

RAISE EXCEPTION 'Suất chiếu mới xung đột với suất chiếu hiện có trong phòng %', NEW.room\_id;

END IF;

END;

END LOOP;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER trigger\_check\_showtime\_gap

BEFORE INSERT

ON public.showtime

FOR EACH ROW

EXECUTE FUNCTION check\_showtime\_gap();

-- Trigger để kiểm tra version\_id trong showtime

CREATE OR REPLACE FUNCTION check\_showtime\_version()

RETURNS TRIGGER AS $$

BEGIN

IF NOT EXISTS (

SELECT 1

FROM public.movie\_version mv

WHERE mv.movie\_id = NEW.movie\_id

AND mv.version\_id = NEW.version\_id

) THEN

RAISE EXCEPTION 'Version % không hợp lệ cho phim %', NEW.version\_id, NEW.movie\_id;

END IF;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER trigger\_check\_showtime\_version

BEFORE INSERT OR UPDATE

ON public.showtime

FOR EACH ROW

EXECUTE FUNCTION check\_showtime\_version();

-- Chỉ mục để tối ưu hiệu suất

CREATE INDEX IF NOT EXISTS idx\_employee\_account\_id ON public.employee(account\_id);

CREATE INDEX IF NOT EXISTS idx\_member\_account\_id ON public.member(account\_id);

CREATE INDEX IF NOT EXISTS idx\_ticket\_invoice ON public.ticket(invoice\_id);

CREATE INDEX IF NOT EXISTS idx\_ticket\_schedule\_seat ON public.ticket(schedule\_seat\_id);

CREATE INDEX IF NOT EXISTS idx\_movie\_review\_movie ON public.movie\_review(movie\_id);

CREATE INDEX IF NOT EXISTS idx\_movie\_review\_account ON public.movie\_review(account\_id);

CREATE INDEX IF NOT EXISTS idx\_movie\_version\_movie\_id ON public.movie\_version(movie\_id);

CREATE INDEX IF NOT EXISTS idx\_movie\_version\_version\_id ON public.movie\_version(version\_id);

CREATE INDEX IF NOT EXISTS idx\_showtime\_version\_id ON public.showtime(version\_id);

CREATE INDEX IF NOT EXISTS idx\_movie\_rating\_code ON public.movie(rating\_code);

-- Insert dữ liệu mẫu

-- 1. movie\_age\_rating

INSERT INTO public.movie\_age\_rating (rating\_code, rating\_name, description) VALUES

('P', 'Mọi lứa tuổi', 'Phim phù hợp với mọi lứa tuổi'),

('T13', '13+', 'Phim dành cho người từ 13 tuổi trở lên'),

('T16', '16+', 'Phim dành cho người từ 16 tuổi trở lên'),

('T18', '18+', 'Phim dành cho người từ 18 tuổi trở lên');

-- 2. roles

INSERT INTO public.roles (role\_name) VALUES

('ADMIN'),

('EMPLOYEE'),

('MEMBER');

-- 3. account

INSERT INTO public.account (username, password, full\_name, email, phone\_number, address, date\_of\_birth, gender, identity\_card, register\_date, status, role\_id) VALUES

('admin1', 'hashed\_password\_1', 'Nguyen Van Admin', 'admin1@example.com', '0901234567', '123 Admin St, Hanoi', '1980-01-01', 'M', '123456789', '2025-06-01', 1, 1),

('employee1', 'hashed\_password\_2', 'Tran Thi Employee', 'employee1@example.com', '0902345678', '456 Employee St, Hanoi', '1990-02-02', 'F', '987654321', '2025-06-01', 1, 2),

('member1', 'hashed\_password\_3', 'Le Van Member', 'member1@example.com', '0903456789', '789 Member St, Hanoi', '1995-03-03', 'M', '456789123', '2025-06-01', 1, 3),

('member2', 'hashed\_password\_4', 'Pham Thi User', 'member2@example.com', '0904567890', '101 User St, HCMC', '1998-04-04', 'F', '654321987', '2025-06-02', 1, 3),

('member3', 'hashed\_password\_5', 'Hoang Van Customer', 'member3@example.com', '0905678901', '202 Customer St, Danang', '1992-05-05', 'M', '789123456', '2025-06-03', 1, 3);

-- 4. employee

INSERT INTO public.employee (account\_id, hire\_date, position, salary) VALUES

(1, '2025-06-01', 'Manager', 20000000),

(2, '2025-06-01', 'Ticket Seller', 10000000);

-- 5. member

INSERT INTO public.member (account\_id, score, tier) VALUES

(3, 100, 'Bạc'),

(4, 1500, 'Vàng'),

(5, 6000, 'Kim cương');

-- 6. cinema\_room

INSERT INTO public.cinema\_room (room\_name, seat\_quantity) VALUES

('Room 1', 100),

('Room 2', 80),

('Room 3', 120),

('Room 4', 60);

-- 7. seat\_type

INSERT INTO public.seat\_type (type\_name, description) VALUES

('Normal', 'Standard seat with basic comfort'),

('VIP', 'Premium seat with extra legroom'),

('Couple', 'Double seat for couples'),

('Premium', 'Luxury seat with reclining feature');

-- 8. seat

INSERT INTO public.seat (room\_id, seat\_row, seat\_column, seat\_type\_id, is\_active) VALUES

(1, 1, 'A', 1, true),

(1, 1, 'B', 1, true),

(1, 1, 'C', 2, true),

(1, 1, 'D', 2, true),

(1, 2, 'A', 1, true),

(1, 2, 'B', 1, true),

(1, 2, 'C', 2, true),

(1, 2, 'D', 2, true),

(1, 3, 'A', 3, true),

(1, 3, 'B', 3, true),

(2, 1, 'A', 1, true),

(2, 1, 'B', 1, true),

(2, 1, 'C', 2, true),

(2, 1, 'D', 2, true),

(2, 2, 'A', 1, true),

(2, 2, 'B', 1, true),

(2, 2, 'C', 3, true),

(2, 2, 'D', 3, true);

-- 9. ticket\_type

INSERT INTO public.ticket\_type (type\_name, description) VALUES

('Adult', 'Ticket for adults'),

('Child', 'Ticket for children under 12'),

('Student', 'Ticket for students with valid ID'),

('Senior', 'Ticket for seniors above 60');

-- 10. version

INSERT INTO public.version (version\_name, description) VALUES

('Phụ đề Tiếng Việt', 'Phim có phụ đề tiếng Việt'),

('Lồng tiếng', 'Phim được lồng tiếng Việt'),

('Phụ đề Tiếng Hàn', 'Phim có phụ đề tiếng Hàn'),

('Phụ đề Tiếng Anh', 'Phim có phụ đề tiếng Anh'),

('3D', 'Phim định dạng 3D');

-- 11. movie

INSERT INTO public.movie (movie\_name, movie\_name\_vn, movie\_name\_en, director, actor, content, duration, production\_company, rating\_code, from\_date, to\_date, large\_image\_url, small\_image\_url, trailer\_url) VALUES

('PHIM ĐIỆN ẢNH DORAEMON: NOBITA VÀ CUỘC PHIÊU LƯU VÀO THẾ GIỚI TRONG TRANH', 'Doraemon: Nobita và cuộc phiêu lưu vào thế giới trong tranh', 'Doraemon: Art Tales', 'Yukiyo Teramoto', 'Wasabi Mizuta, Megumi Ôhara, Yumi Kakazu, Subaru Kimura, Tomokazu Seki', 'Thông qua món bảo bối mới của Doraemon, cả nhóm bạn bước thế giới trong một bức tranh nổi tiếng và bắt gặp cô bạn bí ẩn tên Claire...', 105, NULL, 'P', '2025-05-23', NULL, 'https://iguov8nhvyobj.vcdn.cloud/media/catalog/product/cache/1/image/1800x/71252117777b696995f01934522c402d/c/o/copy\_of\_250220\_dr25\_main\_b1\_localized\_embbed\_1\_.jpg', NULL, 'https://www.youtube.com/watch?v=doraemon\_trailer'),

('NHIỆM VỤ: BẤT KHẢ THI - NGHIỆP BÁO CUỐI CÙNG', 'Nhiệm vụ: Bất khả thi - Nghiệp báo cuối cùng', 'Mission: Impossible - Final Reckoning', 'Christopher McQuarrie', 'Tom Cruise', 'Cuộc đời là tất thảy những lựa chọn. Tom Cruise thủ vai Ethan Hunt trở lại...', 169, NULL, 'T16', '2025-05-30', NULL, 'https://iguov8nhvyobj.vcdn.cloud/media/catalog/product/cache/1/image/c5f0a1eff4c394a251036189ccddaacd/m/i/mi8\_poster\_470x700\_1.jpg', NULL, 'https://www.youtube.com/watch?v=mission\_impossible\_trailer'),

('LILO & STITCH', 'Lilo & Stitch', 'Lilo & Stitch', 'Dean Fleischer Camp', 'Maia Kealoha, Sydney Elizabeth Agudong, Tia Carrere, Chris Sanders', 'Bộ phim live-action Lilo và Stitch đưa câu chuyện kinh điển của Disney năm 2002 trở lại...', 107, NULL, 'P', '2025-05-23', NULL, 'https://iguov8nhvyobj.vcdn.cloud/media/catalog/product/cache/1/image/c5f0a1eff4c394a251036189ccddaacd/c/g/cgv\_350x495-stitch.png', NULL, 'https://www.youtube.com/watch?v=lilo\_stitch\_trailer'),

('YADANG: BA MẶT LẬT KÈO', 'Yadang: Ba mặt lật kèo', 'Yadang', 'HWANG Byeong-gug', 'KANG Ha-neul, YOO Hai-jin, PARK Hae-joon, RYU Kyung-soo, CHAE Won-been', 'Từ giờ trở đi, bạn là kẻ chỉ điểm của tôi...', 122, NULL, 'T18', '2025-05-16', NULL, 'https://iguov8nhvyobj.vcdn.cloud/media/catalog/product/cache/1/image/1800x/71252117777b696995f01934522c402d/t/e/teaser\_poster\_1\_3.jpg', NULL, 'https://www.youtube.com/watch?v=yadang\_trailer'),

('DẾ MÈN: CUỘC PHIÊU LƯU TỚI XÓM LẦY LỘI', 'Dế Mèn: Cuộc phiêu lưu tới xóm lầy lội', 'Cricket: Adventure to Muddy Village', 'Mai Phương', 'Trần Hoàng Sơn, Thái Văn Minh Vũ, Nguyễn Anh Tuấn, Hồ Tiến Đạt, Lê Quang Tuyên', 'Nhà của Dế Mèn tại thành phố ô nhiễm bị phá khiến cậu rời đi tìm nơi trong lành hơn...', 84, NULL, 'P', '2025-05-30', NULL, 'https://iguov8nhvyobj.vcdn.cloud/media/catalog/product/cache/1/image/1800x/71252117777b696995f01934522c402d/d/\_/d\_m\_n\_-\_teaser\_poster\_-\_kthuoc\_social.jpg', NULL, 'https://www.youtube.com/watch?v=cricket\_trailer'),

('MƯỢN HỒN ĐOẠT XÁC', 'Mượn hồn đoạt xác', 'Bring Her Back', 'Danny Philippou, Michael Philippou', 'Billy Barratt, Sally Hawkins, Mischa Heywood', 'Sự trở lại của bộ óc sáng tạo đằng sau Talk to Me...', 103, NULL, 'T18', '2025-05-30', NULL, 'https://iguov8nhvyobj.vcdn.cloud/media/catalog/product/cache/1/image/1800x/71252117777b696995f01934522c402d/b/h/bhb\_16x9\_trailer\_thubmnail.jpg', NULL, 'https://www.youtube.com/watch?v=bring\_her\_back\_trailer'),

('NĂM MƯỜI', 'Năm Mười', 'Fifty', 'Tấn Hoàng Thông', 'Huỳnh Tú Uyên, Trần Vân Anh, Trần Phong, Nam Nam, Vương Thanh Tùng, Hồ Quang Mẫn', 'Lấy cảm hứng từ trò chơi quen thuộc Năm Mười...', 80, NULL, 'T18', '2025-05-30', NULL, 'https://iguov8nhvyobj.vcdn.cloud/media/catalog/product/cache/1/image/1800x/71252117777b696995f01934522c402d/4/0/406x600-510.jpg', NULL, 'https://www.youtube.com/watch?v=fifty\_trailer'),

('MA MÓC HỌNG', 'Ma móc họng', 'Gede', 'Azhar Kinoi Lubis', 'Arla Ailani, Adzana Ashel, Endy Arfian, Raihan Khan', 'Siêu phẩm mùa xuân tại xứ sở Vạn Đảo với hơn 3 triệu vé bán ra...', 98, NULL, 'T16', '2025-05-30', NULL, 'https://iguov8nhvyobj.vcdn.cloud/media/catalog/product/cache/1/image/c5f0a1eff4c394a251036189ccddaacd/g/e/gede\_cgv\_350x495.jpg', NULL, 'https://www.youtube.com/watch?v=gede\_trailer'),

('TỘI ĐỒ', 'Tội đồ', 'Sinners', 'Ryan Coogler', 'Hailee Steinfeld, Jack O''Connell, Michael B. Jordan', 'Mắc kẹt trong cuộc sống thực tại rối ren...', 138, NULL, 'T18', '2025-05-21', NULL, 'https://iguov8nhvyobj.vcdn.cloud/media/catalog/product/cache/1/image/1800x/71252117777b696995f01934522c402d/s/i/sinner.jpg', NULL, 'https://www.youtube.com/watch?v=sinners\_trailer'),

('LẬT MẶT 8: VÒNG TAY NẮNG', 'Lật mặt 8: Vòng tay nắng', 'Face Off 8', 'Lý Hải', 'NSƯT Kim Phương, Long Đẹp Trai, NSƯT Tuyết Thu, Quách Ngọc Tuyên, Đoàn Thế Vinh', 'Một bộ phim về sự khác biệt quan điểm giữa ba thế hệ...', 135, NULL, 'T13', '2025-04-30', NULL, 'https://iguov8nhvyobj.vcdn.cloud/media/catalog/product/cache/1/image/1800x/71252117777b696995f01934522c402d/l/m/lm8\_-\_470x700.jpg', NULL, 'https://www.youtube.com/watch?v=face\_off\_8\_trailer'),

('BÍ KÍP LUYỆN RỒNG', 'Bí kíp luyện rồng', 'How to Train Your Dragon', 'Dean DeBlois', 'Mason Thames, Nico Parker, Gerard Butler', 'Câu chuyện về một chàng trai trẻ với ước mơ trở thành thợ săn rồng...', 100, NULL, 'P', '2025-06-13', NULL, 'https://iguov8nhvyobj.vcdn.cloud/media/catalog/product/cache/1/image/1800x/71252117777b696995f01934522c402d/h/d/hdg\_payoff\_470x700.jpg', NULL, 'https://www.youtube.com/watch?v=how\_to\_train\_your\_dragon\_trailer');

-- 12. movie\_version

INSERT INTO public.movie\_version (movie\_id, version\_id) VALUES

(1, 1), (1, 2),

(2, 1), (2, 3),

(3, 1), (3, 2),

(4, 1),

(5, 2),

(6, 1),

(7, 1), (7, 4),

(8, 1),

(9, 1),

(10, 1), (10, 4),

(11, 1), (11, 2);

-- 13. type

INSERT INTO public.type (type\_name) VALUES

('Hoạt Hình'),

('Phiêu Lưu'),

('Hành Động'),

('Hồi hộp'),

('Gia đình'),

('Hài'),

('Kinh Dị'),

('Tâm Lý'),

('Thần thoại'),

('Lịch Sử'),

('Khoa Học Viễn Tưởng'),

('Tình cảm'),

('Bí ẩn');

-- 14. movie\_type

INSERT INTO public.movie\_type (movie\_id, type\_id) VALUES

(1, 1), (1, 2),

(2, 3), (2, 4), (2, 2),

(3, 5), (3, 6), (3, 2),

(4, 3),

(5, 1),

(6, 7),

(7, 7),

(8, 7),

(9, 4), (9, 7), (9, 8),

(10, 5), (10, 8),

(11, 6), (11, 3), (11, 2), (11, 9);

-- 15. movie\_review

INSERT INTO public.movie\_review (movie\_id, account\_id, rating, comment, review\_date) VALUES

(1, 3, 4, 'Phim rất dễ thương, phù hợp cho gia đình!', '2025-06-06 16:00:00'),

(2, 4, 5, 'Hành động đỉnh cao, Tom Cruise không làm mình thất vọng!', '2025-06-06 22:00:00'),

(3, 5, 3, 'Bản live-action khá ổn nhưng không bằng bản hoạt hình.', '2025-06-07 16:30:00'),

(5, 3, 4, 'Phim Việt Nam rất sáng tạo, đáng xem!', '2025-06-08 23:00:00');

-- 16. promotion

INSERT INTO public.promotion (title, detail, discount\_level, start\_time, end\_time, image\_url, is\_active) VALUES

('Summer Sale', 'Get 20% off on all tickets', 20.00, '2025-06-01 00:00:00', '2025-06-30 23:59:59', 'https://iguov8nhvyobj.vcdn.cloud/media/wysiwyg/2025/062025/240x201\_7\_.jpg', true),

('VIP Discount', '10% off for VIP seats', 10.00, '2025-06-01 00:00:00', '2025-06-15 23:59:59', 'https://iguov8nhvyobj.vcdn.cloud/media/wysiwyg/2025/062025/240x201\_4\_.png', true),

('Family Package', 'Buy 3 tickets, get 1 free', 25.00, '2025-06-05 00:00:00', '2025-06-20 23:59:59', 'https://iguov8nhvyobj.vcdn.cloud/media/wysiwyg/2025/052025/MMP\_Promote\_240x201.jpg', true),

('Student Discount', '15% off for students with valid ID', 15.00, '2025-06-10 00:00:00', '2025-06-30 23:59:59', 'https://iguov8nhvyobj.vcdn.cloud/media/wysiwyg/2025/052025/240x201\_4\_.jpg', true);

-- 17. movie\_promotion

INSERT INTO public.movie\_promotion (movie\_id, promotion\_id) VALUES

(1, 1),

(2, 2),

(3, 3),

(5, 1);

-- 18. show\_dates

INSERT INTO public.show\_dates (show\_date) VALUES

('2025-06-06'),

('2025-06-07'),

('2025-06-08'),

('2025-06-09');

-- 19. schedule

INSERT INTO public.schedule (schedule\_time) VALUES

('14:30'),

('17:00'),

('19:00'),

('21:30');

-- 20. showtime

INSERT INTO public.showtime (movie\_id, show\_date\_id, schedule\_id, room\_id, version\_id, available\_seats, room\_number) VALUES

(1, 1, 1, 1, 1, 100, 'R1'),

(1, 1, 2, 1, 2, 100, 'R1'),

(2, 1, 3, 2, 1, 80, 'R2'),

(3, 2, 1, 3, 1, 120, 'R3'),

(4, 2, 2, 4, 1, 60, 'R4'),

(5, 3, 4, 1, 2, 100, 'R1'),

(11, 4, 4, 2, 1, 80, 'R2');

-- 21. schedule\_seat

INSERT INTO public.schedule\_seat (showtime\_id, seat\_id, seat\_price, status) VALUES

(1, 1, 80000, 1),

(1, 2, 80000, 1),

(1, 3, 120000, 0),

(1, 4, 120000, 0),

(4, 5, 120000, 1),

(4, 6, 120000, 0),

(6, 7, 75000, 1),

(6, 8, 75000, 0),

(2, 9, 100000, 0),

(7, 10, 75000, 0);

-- 22. invoice

INSERT INTO public.invoice (account\_id, booking\_date, total\_amount, payment\_method, use\_score, add\_score, status, movie\_name, seat\_number) VALUES

(3, '2025-06-06 10:00:00', 140000, 'online', 0, 20, 1, 'Doraemon: Nobita và cuộc phiêu lưu vào thế giới trong tranh', '1A, 1B'),

(4, '2025-06-07 12:00:00', 120000, 'online', 0, 10, 1, 'Lilo & Stitch', '1C'),

(5, '2025-06-08 15:00:00', 75000, 'credit\_card', 50, 15, 1, 'Dế Mèn: Cuộc phiêu lưu tới xóm lầy lội', '2A');

-- 23. ticket

INSERT INTO public.ticket (invoice\_id, schedule\_seat\_id, ticket\_type\_id, price) VALUES

(1, 1, 1, 80000),

(1, 2, 2, 60000),

(2, 5, 1, 120000),

(3, 7, 3, 75000);

-- 24. invoice\_promotion

INSERT INTO public.invoice\_promotion (invoice\_id, promotion\_id) VALUES

(1, 1),

(2, 3),

(3, 1);

COMMIT;