This project models the scenario in SQL, ensuring the schema follows 1NF, 2NF, 3NF, and BCNF normalization principles.

Description of BookMyShow Database Design

1. Entities and Their Attributes

a) Theatre

- Attributes:

- `theatre\_id` (Primary Key): Unique identifier for each theatre.

- `name`: Name of the theatre.

- `address`: Full address/location of the theatre.

- `contact\_number`: Phone number for the theatre.

b) Screen

- Attributes:

- `screen\_id` (Primary Key): Unique identifier for each screen in a theatre.

- `theatre\_id` (Foreign Key): Identifies which theatre the screen belongs to.

- `screen\_name`: Name or number label of the screen (e.g., "Screen 1").

- `screen\_type`: Type of screen (Standard, IMAX, Playhouse, etc.).

c) Movie

- Attributes:

- `movie\_id` (Primary Key): Unique identifier for each movie.

- `title`: Name of the movie.

- `language`: Language in which the movie is shown.

- `duration`: Duration in minutes.

- `censor\_rating`: Movie certification/rating (e.g., UA, A).

d) Format

- Attributes:

- `format\_id` (Primary Key): Unique identifier for format/technology type.

- `format\_name`: Format/technology name (2D, 3D, 4K Dolby 7.1, etc.).

e) Show

- Attributes:

- `show\_id` (Primary Key): Unique identifier for each show.

- `movie\_id` (Foreign Key): The movie being shown.

- `theatre\_id` (Foreign Key): Theatre where the show is scheduled.

- `screen\_id` (Foreign Key): The screen in the theatre.

- `format\_id` (Foreign Key): The format/technology for the show.

- `show\_date`: Date of the show.

- `show\_time`: Time of the show.

- `price`: Ticket price for that show.

2. Relationships

- A Theatre has multiple Screens.

- A Screen belongs to one Theatre.

- A Movie can have multiple Shows.

- A Show is associated with one Movie, one Theatre, one Screen, one Format, at a particular Date & Time, and has a Price.

3. Table Description

Each entity becomes a table in the database, with columns for its attributes. Foreign key relationships enforce data integrity between connected tables.

4. SQL Summary

- The CREATE TABLE statements set up structured, relational tables that capture all necessary details and relationships for the movie booking platform.

- The INSERT statements provide initial sample data for each entity.

- The SELECT query at the end fetches all show details for a specific theatre and date, with format and screen info included.

-- BookMyShow SQL Case Study

-- Theatre

CREATE TABLE Theatre (

theatre\_id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(100) NOT NULL,

address VARCHAR(255),

contact\_number VARCHAR(15)

);

-- Screen

CREATE TABLE Screen (

screen\_id INT AUTO\_INCREMENT PRIMARY KEY,

theatre\_id INT,

screen\_name VARCHAR(50),

screen\_type VARCHAR(50),

FOREIGN KEY (theatre\_id) REFERENCES Theatre(theatre\_id)

);

-- Movie

CREATE TABLE Movie (

movie\_id INT AUTO\_INCREMENT PRIMARY KEY,

title VARCHAR(100) NOT NULL,

language VARCHAR(50),

duration INT,

censor\_rating VARCHAR(10)

);

-- Format

CREATE TABLE Format (

format\_id INT AUTO\_INCREMENT PRIMARY KEY,

format\_name VARCHAR(50) NOT NULL

);

--Show

CREATE TABLE Show (

show\_id INT AUTO\_INCREMENT PRIMARY KEY,

movie\_id INT,

theatre\_id INT,

screen\_id INT,

format\_id INT,

show\_date DATE,

show\_time TIME,

price DECIMAL(10,2),

FOREIGN KEY (movie\_id) REFERENCES Movie(movie\_id),

FOREIGN KEY (theatre\_id) REFERENCES Theatre(theatre\_id),

FOREIGN KEY (screen\_id) REFERENCES Screen(screen\_id),

FOREIGN KEY (format\_id) REFERENCES Format(format\_id)

);

-- Theatre

INSERT INTO Theatre (name, address, contact\_number)

VALUES ('PVR: Nexus Forum', ‘Kukatpally,Hyderabad, Telangana', '7330991292');

-- Screens

INSERT INTO Screen (theatre\_id, screen\_name, screen\_type)

VALUES (1, 'Screen 1', 'Standard'),

(1, 'Screen 2', 'IMAX');

-- Movies

INSERT INTO Movie (title, language, duration, censor\_rating)

VALUES (‘Coolie’, 'Telugu', 150, 'A'),

(‘War 2', 'Hindi', 160, 'UA'),

('Nobody 2, 'English', 190, 'UA');

-- Format

INSERT INTO Format (format\_name)

VALUES ('2D'), ('3D'), ('4K Dolby 7.1');

-- Shows

INSERT INTO Show (movie\_id, theatre\_id, screen\_id, format\_id, show\_date, show\_time, price)

VALUES

(1, 1, 1, 1, '2025-08-14', '10:45:00', 250.00),

(2, 1, 1, 3, '2025-08-14', '14:30:00', 300.00),

(2, 1, 1, 3, '2025-08-14', '20:45:00', 300.00),

(3, 1, 2, 2, '2025-08-14', '19:20:00', 300.00);

--P2 Requirements

SELECT

m.title AS movie\_name,

sh.show\_time,

f.format\_name,

sc.screen\_name,

sh.price

FROM Show sh

JOIN Movie m ON sh.movie\_id = m.movie\_id

JOIN Theatre t ON sh.theatre\_id = t.theatre\_id

JOIN Screen sc ON sh.screen\_id = sc.screen\_id

JOIN Format f ON sh.format\_id = f.format\_id

WHERE t.name = 'PVR: Nexus Forum'

AND sh.show\_date = '2025-08-14'

ORDER BY m.title, sh.show\_time;