**Problem Solving Case: Bookmyshow**

**Problem 1 -** As part of this assignment, we need to list down all the entities, their attributes and the table structures for the scenario mentioned in the previous slide. You also need to write the SQL queries required to create these tables along with a few sample entries. Ensure the tables follow 1NF, 2NF, 3NF and BCNF rules.

**Entities -**

1. Theater
2. Movie
3. Show
4. Booking
5. User

**Attributes -**

1. **Theater:**

Attributes - TheaterID (Primary Key), TheaterName, Location

1. **Movie:**

Attributes - MovieID (Primary Key), MovieName, Genre, Language

1. **Show:**

Attributes - ShowID (Primary Key), TheatreID (Foreign Key), MovieID (Foreign Key), ShowDate, ShowTime

1. **Booking:**

Attributes - BookingID (Primary Key), ShowID (Foreign Key), UserID, BookingDate, TotalSeats, Amount

1. **User:**

Attributes - UserID (Primary Key), UserName, Email, Phone

**Table Structures With SQL Queries -**

1. **Theatre Table -**

CREATE TABLE Theatre (

TheatreID INT PRIMARY KEY,

TheatreName VARCHAR(255) NOT NULL,

Location VARCHAR(255) NOT NULL

);

1. **Movie Table -**

CREATE TABLE Movie (

MovieID INT PRIMARY KEY,

MovieName VARCHAR(255) NOT NULL,

Genre VARCHAR(255),

Language VARCHAR(50)

);

1. **Show Table -**

CREATE TABLE Show (

ShowID INT PRIMARY KEY,

TheatreID INT,

MovieID INT,

ShowDate DATE,

ShowTime TIME,

FOREIGN KEY (TheatreID) REFERENCES Theatre(TheatreID),

FOREIGN KEY (MovieID) REFERENCES Movie(MovieID)

);

1. **User Table -**

CREATE TABLE User (

UserID INT PRIMARY KEY,

UserName VARCHAR(255) NOT NULL,

Email VARCHAR(255) UNIQUE NOT NULL,

Phone VARCHAR(20) UNIQUE NOT NULL

);

1. **Booking Table -**

CREATE TABLE Booking (

BookingID INT PRIMARY KEY,

ShowID INT,

UserID INT,

BookingDate DATE,

TotalSeats INT,

Amount Decimal(10, 2),

FOREIGN KEY (ShowID) REFERENCES Show(ShowID),

FOREIGN KEY (UserID) REFERENCES User(UserID)

);

**SQL Queries To Add Some Sample Data -**

1. **Theatre Table -**

INSERT INTO Theatre (TheatreID, TheatreName, Location) VALUES

(1, ‘INOX Theatre’, ‘Pune’),

(2, ‘Natraj Theatre’, ‘Jhansi’);

1. **Movie Table -**

INSERT INTO Movie (MovieID, MovieName, Genre, Language) VALUES

(91, 'Avengers', 'Action', 'English'),

(92, 'Angry Birds', 'Comedy', 'Hindi');

1. **Show Table -**

INSERT INTO Show (ShowID, TheatreID, MovieID, ShowDate, ShowTime) VALUES

(1001, 1, 101, '2024-02-01', '18:00:00'),

(1002, 1, 102, '2024-02-02', '20:30:00'),

(1003, 2, 101, '2024-02-03', '15:45:00');

1. **User Table -**

INSERT INTO User (UserID, UserName, Email, Phone) VALUES

(10001, 'Siddharth Soni', 'sid@gmail.com', '+1234567890'),

(10002, 'Sunny Yadav', 'sunny@gmail.com', '+9876543210');

1. **Booking Table -**

INSERT INTO Booking (BookingID, ShowID, UserID, BookingDate, TotalSeats, Amount) VALUES

(5001, 1001, 10001, '2024-02-01', 2, 200.00),

(5002, 1002, 10002, '2024-02-02', 3, 300.50),

(5003, 1003, 10001, '2024-02-03', 1, 150.00);

**Problem 2 -** Write a query to list down all the shows on a given date at a given theater along with their respective show timings.

SELECT

Theatre.TheatreName, Movie.MovieName,

Show.ShowDate, Show.ShowTime

FROM Show

JOIN Theatre ON Show.TheatreID = Theatre.TheatreID

JOIN Movie ON Show.MovieID = Movie.MovieID

WHERE Show.ShowDate = '2024-02-01'

AND Theatre.TheatreID = 1;