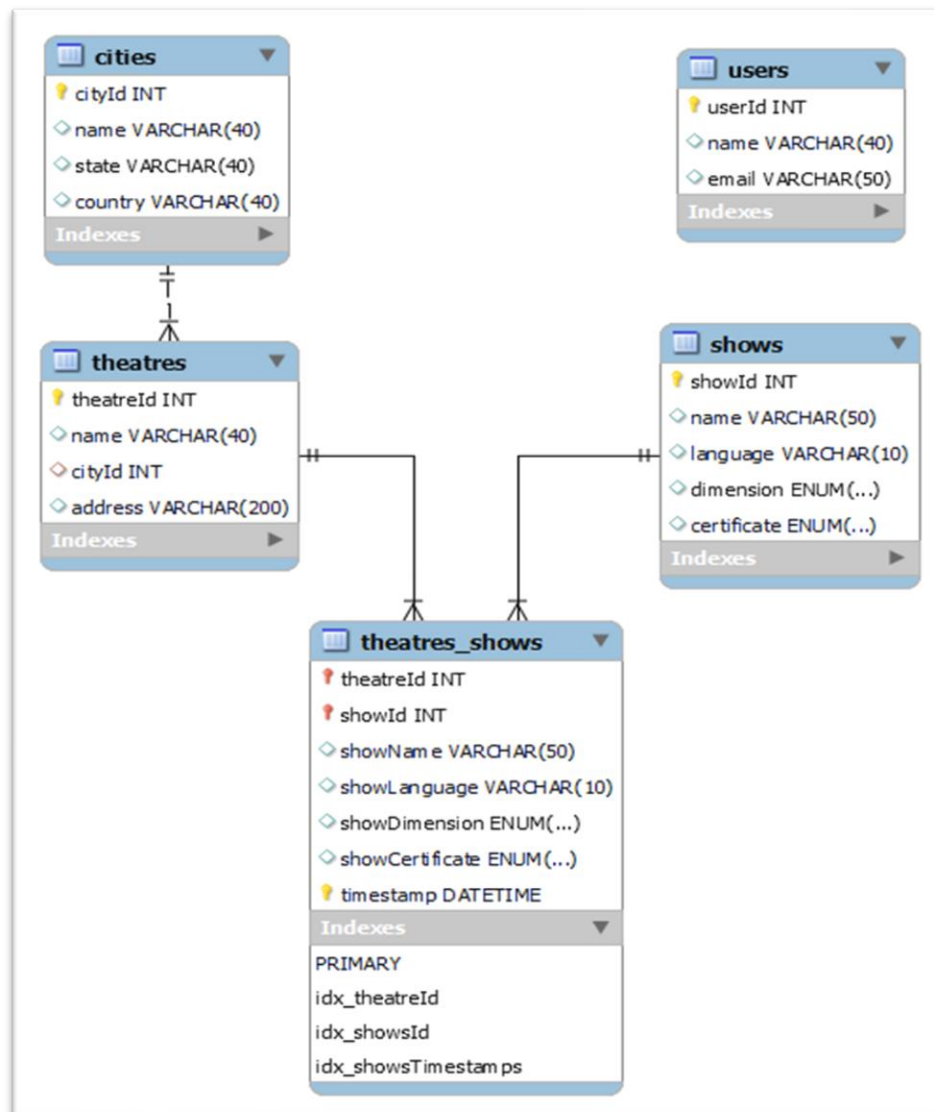


Schema:

Tables :

1. *Users*
2. *Cities*
3. *Theatres*
4. *Shows*
5. *Theatres_shows*



Considerations

1. Flexibility and scalability is given priority over some redundant columns hence the use of foreign key constraints is avoided. Still, foreign key references are used to maintain the data integrity.
2. The 'Theatre-wise shows and timings on a given date' query has been given higher priority, as this will be the most used operation hence database schema is designed around this. For this, there is a theatres-shows-timings table just to serve this query but it is consistent with the theatre and shows table using references.
3. Because of the separate table there is redundant data but it avoids joins for most used queries, hence reducing read time, and making the queries less complex and straightforward.
4. Indexing is performed on theatreId, and showId columns of 'theatres_shows' table separately to support faster queries.

Results

1. Shows in a given theatre on a given date

```
SELECT
    showName AS Show_Name,
    timestamp AS Show_Timing,
    showLanguage,
    showDimension,
    showCertificate

FROM
    theatres_shows
WHERE
    DATE(timestamp) = '2023-09-10' -- Replace with your desired date
    AND theatreId = 1; -- Replace with the desired theater's ID
```

<

2. Upcoming 7 days shows in a given theatre and date

theatres_shows ×

Search results						
Theatre_Id int	Show_Name varchar	Show_Timing datetime	showLanguage varchar(10)	showDimension enum('2D','3D','IM')	showCertificate enum('UA','U','A','S')	
1	Bahubali: The Beginning	2023-09-10 15:00:00	Hindi	2D	UA	
2	Dangal	2023-09-10 15:30:00	Hindi	2D	U	
3	Kabali	2023-09-10 19:00:00	Tamil	2D	UA	
4	Bahubali: The Beginning	2023-09-11 15:00:00	Hindi	2D	UA	
5	Dangal	2023-09-11 15:30:00	Hindi	2D	U	
6	Kabali	2023-09-11 19:00:00	Tamil	2D	UA	
7	Bahubali: The Beginning	2023-09-12 15:00:00	Hindi	2D	UA	
8	Dangal	2023-09-12 15:30:00	Hindi	2D	U	
9	Kabali	2023-09-12 19:00:00	Tamil	2D	UA	
10	Bahubali: The Beginning	2023-09-13 15:00:00	Hindi	2D	UA	
11	Dangal	2023-09-13 15:30:00	Hindi	2D	U	
12	Kabali	2023-09-13 19:00:00	Tamil	2D	UA	
13	Bahubali: The Beginning	2023-09-14 15:00:00	Hindi	2D	UA	
14	Dangal	2023-09-14 15:30:00	Hindi	2D	U	
15	Kabali	2023-09-14 19:00:00	Tamil	2D	UA	

All Queries:

```
USE bookmyshow;

CREATE TABLE users (
    userId INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(40),
    email VARCHAR(50)
);

CREATE TABLE cities (
    cityId INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(40),
    state VARCHAR(40),
    country VARCHAR(40)
);

CREATE TABLE shows (
    showId INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(50),
    language VARCHAR(10),
    dimension ENUM('2D', '3D', 'IMAX'),
);

CREATE TABLE theatres (
    theatreId INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(40),
    cityId INT ,
    address VARCHAR(200),
    FOREIGN KEY (cityId) REFERENCES cities(cityId) ON UPDATE CASCADE
);

CREATE TABLE theatres_shows(
    theatreId INT,
    showId INT,
    showName varchar(50),
    showLanguage VARCHAR(10),
    showDimension ENUM( '2D', '3D', 'IMAX'),
    showCertificate ENUM('UA', 'U', 'A', 'S'),
    timestamp DATETIME,
    PRIMARY KEY (theatreId, showId, timestamp),
    FOREIGN KEY (theatreId) REFERENCES theatres(theatreId) ON UPDATE CASCADE,
    FOREIGN key (showId) REFERENCES shows(showId) ON UPDATE CASCADE
);

ALTER Table shows
CHANGE dimation dimension ENUM('2D', '3D', 'IMAX');
```

```

ALTER Table shows
ADD certificate ENUM('UA', 'U', 'A', 'S')
-- query for: all the shows on a given date at a given theatre along with
their respective show timings.
SELECT
    showName AS Show_Name,
    timestamp AS Show_Timing,
    showLanguage,
    showDimension,
    showCertificate

FROM
    theatres_shows
WHERE
    DATE(timestamp) = '2023-09-10' -- Replace with your desired date
    AND theatreId = 1; -- Replace with the desired theater's ID

-- indexing for optimization

CREATE INDEX idx_theatreIds ON theatres_shows(theatreId);

CREATE INDEX idx_showsIds ON theatres_shows(showId);

create INDEX idx_showsTimestamp on theatres_shows(showId, timestamp);

-- for next seven days

SELECT
    theatreId AS Theatre_Id,
    showName AS Show_Name,
    timestamp AS Show_Timing,
    showLanguage,
    showDimension,
    showCertificate
FROM
    theatres_shows
WHERE
    DATE(timestamp) BETWEEN '2023-09-10' AND DATE_ADD('2023-09-10', INTERVAL 7
DAY)
    AND theatreId = 1
    ORDER BY timestamp
    LIMIT 15 OFFSET 0 ;

```

Data Insert queries

```
-- Inserting data into the 'users' table
INSERT INTO users (name, email) VALUES
    ('John Doe', 'johndoe@example.com'),
    ('Rahul Mishra', 'rahulmishra@example.com'),
    ('Jason Marz', 'jasonmarz@example.com');

-- Inserting data into the 'cities' table
INSERT INTO cities (name, state, country) VALUES
    ('Mumbai', 'Maharashtra', 'India'),
    ('Delhi', 'Delhi', 'India'),
    ('Chennai', 'Tamil Nadu', 'India'),
    ('Hyderabad', 'Telangana', 'India');

-- Inserting data into the 'shows' table
INSERT INTO shows (name, language, dimension) VALUES
    ('Bahubali: The Beginning', 'Hindi', '2D'),
    ('Dangal', 'Hindi', '2D'),
    ('Kabali', 'Tamil', '2D'),
    ('PK', 'Hindi', '3D'),
    ('Baahubali 2: The Conclusion', 'Telugu', 'IMAX');

-- Inserting data into the 'theatres' table
INSERT INTO theatres (name, cityId, address) VALUES
    ('PVR Cinemas Andheri', 1, '123 Main Street, Andheri West, Mumbai'),
    ('INOX Cinemas Connaught Place', 2, '456 Central Street, Connaught Place, Delhi'),
    ('Sathyam Cinemas', 3, '789 Mount Road, Chennai'),
    ('Cinepolis Hyderabad', 4, '101 Hi-Tech City, Hyderabad');

-- Inserting data into the 'theatres_shows' table
INSERT INTO theatres_shows (theatreId, showId, showName, showLanguage, showDimension, showCertificate, timestamp) VALUES
    -- Bahubali: The Beginning Showings
    (1, 1, 'Bahubali: The Beginning', 'Hindi', '2D', 'UA', '2023-09-10 15:00:00'),
    (1, 1, 'Bahubali: The Beginning', 'Hindi', '2D', 'UA', '2023-09-11 15:00:00'),
    (1, 1, 'Bahubali: The Beginning', 'Hindi', '2D', 'UA', '2023-09-12 15:00:00'),
    (1, 1, 'Bahubali: The Beginning', 'Hindi', '2D', 'UA', '2023-09-13 15:00:00'),
    (1, 1, 'Bahubali: The Beginning', 'Hindi', '2D', 'UA', '2023-09-14 15:00:00'),
    (1, 1, 'Bahubali: The Beginning', 'Hindi', '2D', 'UA', '2023-09-15 15:00:00'),
```

```

(1, 1, 'Bahubali: The Beginning', 'Hindi', '2D', 'UA', '2023-09-16
15:00:00'),
(1, 1, 'Bahubali: The Beginning', 'Hindi', '2D', 'UA', '2023-09-17
15:00:00'),
(1, 1, 'Bahubali: The Beginning', 'Hindi', '2D', 'UA', '2023-09-18
15:00:00'),
(1, 1, 'Bahubali: The Beginning', 'Hindi', '2D', 'UA', '2023-09-19
15:00:00'),
-- Dangal Showings
(1, 2, 'Dangal', 'Hindi', '2D', 'U', '2023-09-10 15:30:00'),
(1, 2, 'Dangal', 'Hindi', '2D', 'U', '2023-09-11 15:30:00'),
(1, 2, 'Dangal', 'Hindi', '2D', 'U', '2023-09-12 15:30:00'),
(1, 2, 'Dangal', 'Hindi', '2D', 'U', '2023-09-13 15:30:00'),
(1, 2, 'Dangal', 'Hindi', '2D', 'U', '2023-09-14 15:30:00'),
(1, 2, 'Dangal', 'Hindi', '2D', 'U', '2023-09-15 15:30:00'),
(1, 2, 'Dangal', 'Hindi', '2D', 'U', '2023-09-16 15:30:00'),
(1, 2, 'Dangal', 'Hindi', '2D', 'U', '2023-09-17 15:30:00'),
(1, 2, 'Dangal', 'Hindi', '2D', 'U', '2023-09-18 15:30:00'),
(1, 2, 'Dangal', 'Hindi', '2D', 'U', '2023-09-19 15:30:00'),
-- Kabali Showings
(1, 3, 'Kabali', 'Tamil', '2D', 'UA', '2023-09-10 19:00:00'),
(1, 3, 'Kabali', 'Tamil', '2D', 'UA', '2023-09-11 19:00:00'),
(1, 3, 'Kabali', 'Tamil', '2D', 'UA', '2023-09-12 19:00:00'),
(1, 3, 'Kabali', 'Tamil', '2D', 'UA', '2023-09-13 19:00:00'),
(1, 3, 'Kabali', 'Tamil', '2D', 'UA', '2023-09-14 19:00:00'),
(1, 3, 'Kabali', 'Tamil', '2D', 'UA', '2023-09-15 19:00:00'),
(1, 3, 'Kabali', 'Tamil', '2D', 'UA', '2023-09-16 19:00:00'),
(1, 3, 'Kabali', 'Tamil', '2D', 'UA', '2023-09-17 19:00:00'),
(1, 3, 'Kabali', 'Tamil', '2D', 'UA', '2023-09-18 19:00:00'),
(1, 3, 'Kabali', 'Tamil', '2D', 'UA', '2023-09-19 19:00:00'),
-- PK Showings
(2, 4, 'PK', 'Hindi', '3D', 'UA', '2023-09-10 14:15:00'),
(2, 4, 'PK', 'Hindi', '3D', 'UA', '2023-09-10 17:15:00'),
(2, 4, 'PK', 'Hindi', '3D', 'UA', '2023-09-11 14:15:00'),
(2, 4, 'PK', 'Hindi', '3D', 'UA', '2023-09-11 17:15:00'),
(2, 4, 'PK', 'Hindi', '3D', 'UA', '2023-09-12 14:15:00'),
(2, 4, 'PK', 'Hindi', '3D', 'UA', '2023-09-12 17:15:00'),
(2, 4, 'PK', 'Hindi', '3D', 'UA', '2023-09-13 14:15:00'),
(2, 4, 'PK', 'Hindi', '3D', 'UA', '2023-09-13 17:15:00'),
(2, 4, 'PK', 'Hindi', '3D', 'UA', '2023-09-14 14:15:00'),
(2, 4, 'PK', 'Hindi', '3D', 'UA', '2023-09-14 17:15:00'),
-- Baahubali 2 Showings
(4, 5, 'Baahubali 2: The Conclusion', 'Telugu', 'IMAX', 'U', '2023-09-10
20:30:00'),
(4, 5, 'Baahubali 2: The Conclusion', 'Telugu', 'IMAX', 'U', '2023-09-10
22:00:00'),
(4, 5, 'Baahubali 2: The Conclusion', 'Telugu', 'IMAX', 'U', '2023-09-11
20:30:00'),

```

```
(4, 5, 'Baahubali 2: The Conclusion', 'Telugu', 'IMAX', 'U', '2023-09-11  
22:00:00'),  
(4, 5, 'Baahubali 2: The Conclusion', 'Telugu', 'IMAX', 'U', '2023-09-12  
20:30:00'),  
(4, 5, 'Baahubali 2: The Conclusion', 'Telugu', 'IMAX', 'U', '2023-09-12  
22:00:00'),  
(4, 5, 'Baahubali 2: The Conclusion', 'Telugu', 'IMAX', 'U', '2023-09-13  
20:30:00'),  
(4, 5, 'Baahubali 2: The Conclusion', 'Telugu', 'IMAX', 'U', '2023-09-13  
22:00:00'),  
(4, 5, 'Baahubali 2: The Conclusion', 'Telugu', 'IMAX', 'U', '2023-09-14  
20:30:00'),  
(4, 5, 'Baahubali 2: The Conclusion', 'Telugu', 'IMAX', 'U', '2023-09-14  
22:00:00');
```