UE21CS351A: DATABASE MANAGEMENT SYSTEM

MINIPROJECT FINAL REPORT

PROJECT TITLE: CONCERT TICKET BOOKING SYSTEM

TEAM MEMBERS:

RIA R KULKARNI – PES1UG21CS487

RIYA JAYAKUMAR – PES1UG21CS492

5TH SEMESTER, 'H' SECTION

TABLE OF CONTENTS

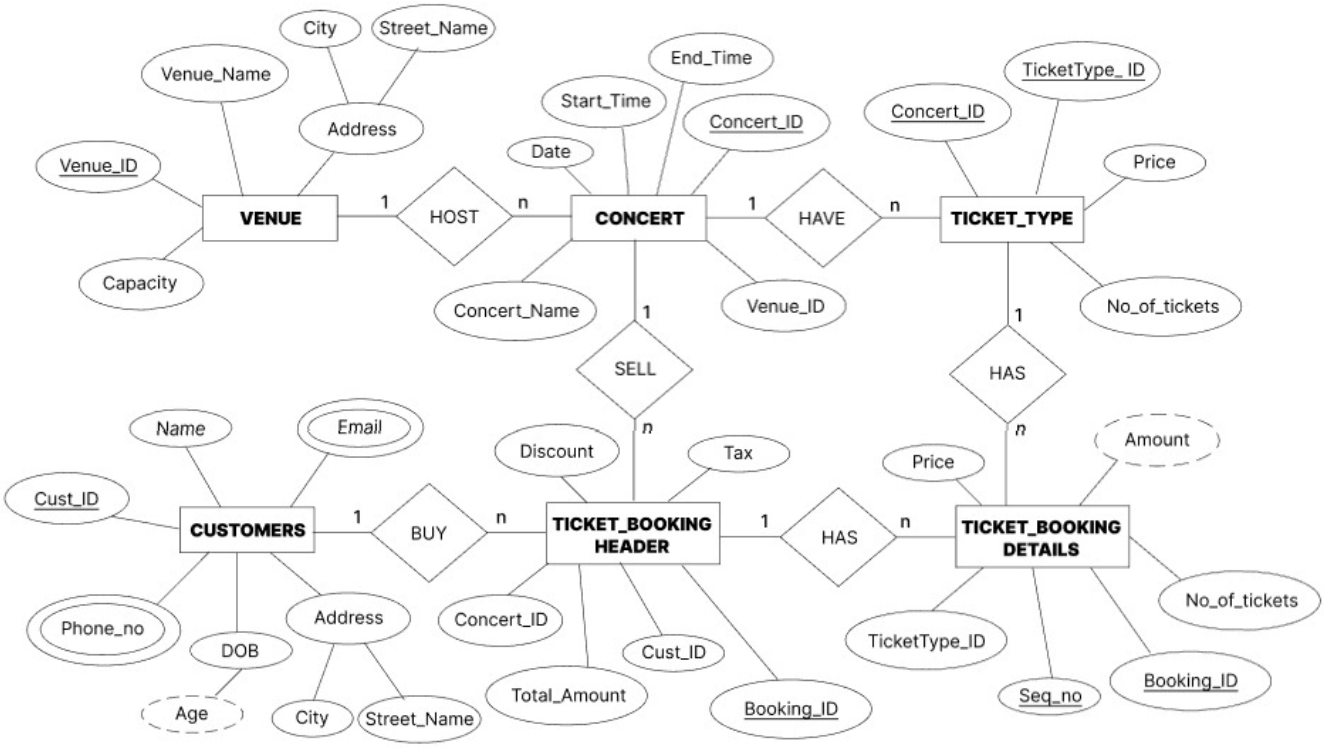
PROJECT ABSTRACT

Concert Ticket Booking System is an online platform designed to streamline the process of booking tickets for various music concerts and events. The system aims to provide a user-friendly interface for customers to browse, select, and purchase tickets for their favourite concerts, while also offering event organizers and administrators a robust platform to manage events, venues, and ticket sales efficiently. The system contributes to the growth of the entertainment industry by connecting fans with their favourite artists and creating memorable concert experiences.

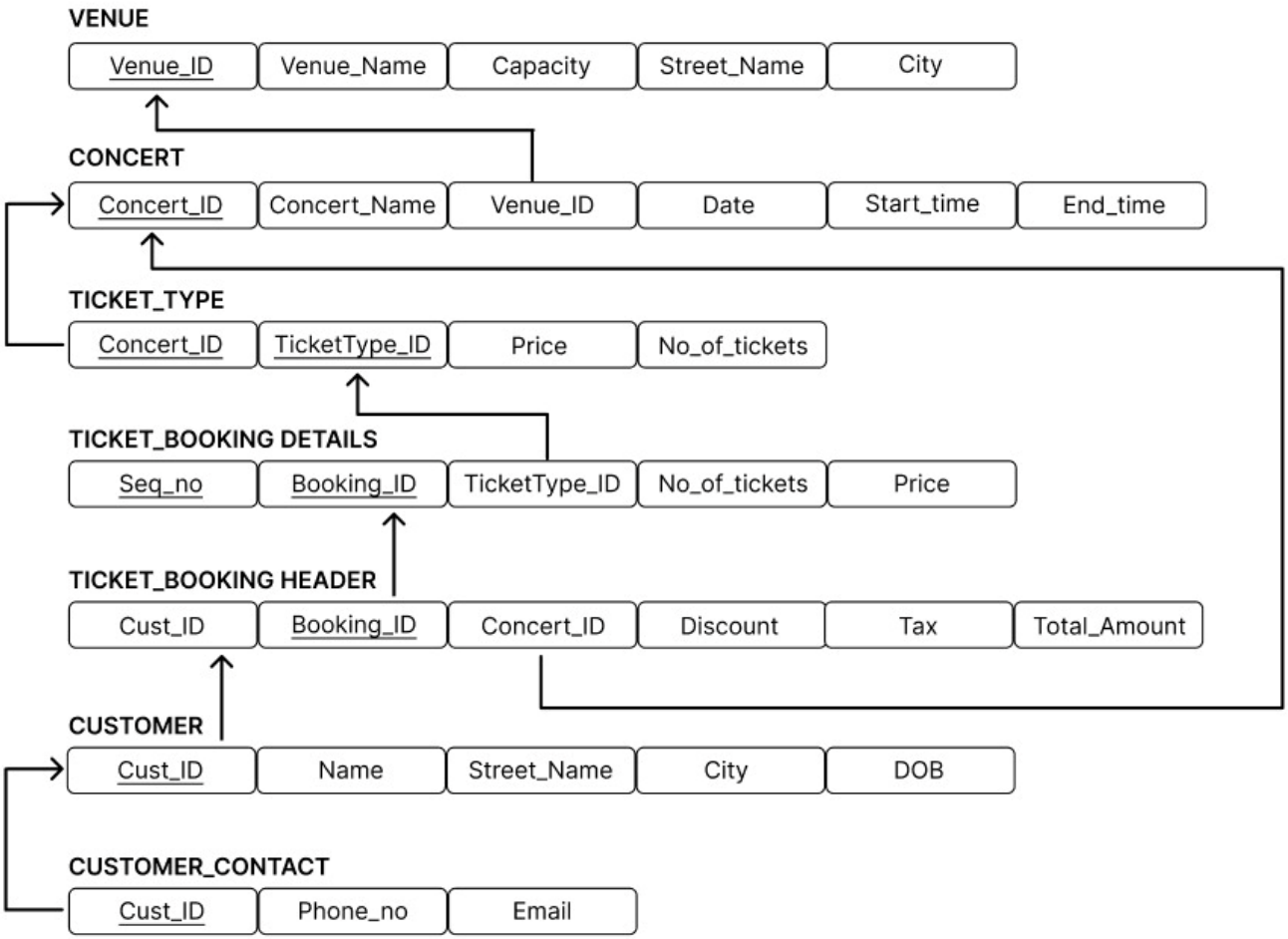
In the dynamic and competitive world of live events, managing revenue effectively is paramount to the success of both event organizers and venues. A revenue management system for concert ticket bookings is a sophisticated solution designed to optimize ticket pricing, enhance customer experience, and maximize revenue streams. This system leverages data analytics, pricing strategies, and customer behaviour analysis to ensure that concert organizers capitalize on every opportunity.

Thus, in today's time, where consumer behaviour and market trends are constantly evolving, the need for a sophisticated revenue management system for concert ticket bookings has never been more crucial and thus, we've implemented a Concert Ticket Booking System that incorporates revenue tracking, using tools such as MySQL and PHP for backend and frontend respectively.

ER DIAGRAM


City
Street_Name
Venue_Name
Address
Venue_ID
VENUE
Capacity
1
HOST
n
Date
Start_Time
End_Time
Concert_ID
CONCERT
Concert_Name
Venue_ID
1
HAVE
n
Concert_ID
TicketType_ ID
Price
TICKET_TYPE
No_of_tickets
1
SELL
n
1
HAS
n
Name
Email
Cust_ID
CUSTOMERS
Phone_no
DOB
Age
Address
City
Street_Name
1
BUY
n
Discount
Tax
TICKET_BOOKING HEADER
Concert_ID
Cust_ID
Total_Amount
Booking_ID
1
HAS
n
Price
Amount
TICKET_BOOKING DETAILS
TicketType_ID
Seq_no
Booking_ID
No_of_tickets

RELATIONAL SCHEMA



DDL SQL COMMANDS

1) Creating “customer” table:

```
CREATE TABLE customer(
     cust_id VARCHAR(20) NOT NULL,
     cust_name VARCHAR(20),
     email VARCHAR(50),
     phone_no VARCHAR(12),
    street_name VARCHAR(30),
     city VARCHAR(30),
    DOB date,
    age INT(5),
     PRIMARY KEY (cust_id)
);
```

---

2) Creating “venues” table:

```
CREATE TABLE venues(
     venue_id VARCHAR(20) NOT NULL,
    venue_name VARCHAR(30),
     capacity INT,
    city VARCHAR(30),
    street_name VARCHAR(30),
     CHECK (capacity>=0),
     PRIMARY KEY (venue_id)
);
```

---

3) Creating “concert” table:

```
CREATE TABLE concert (
    concert_id VARCHAR(20) NOT NULL,
    venue_id VARCHAR(20),
    concert_name VARCHAR(30),
    date_of_concert DATE,
    start_time TIME,
    end_time TIME,
    PRIMARY KEY (concert_id),
```

```
    FOREIGN KEY (venue_id) REFERENCES venues(venue_id) ON UPDATE CASCADE
ON DELETE NO ACTION
);
```

---

4) Creating “ticket_type” table:

```
CREATE TABLE ticket_type(
    ticketType_id VARCHAR(20) NOT NULL,
    concert_id VARCHAR(20) NOT NULL,
    price FLOAT,
    tickets_available INT,
    CHECK (price >= 0 AND tickets_available >= 0),
    PRIMARY KEY (ticketType_id, concert_id),
    FOREIGN KEY (concert_id) REFERENCES concert(concert_id) ON UPDATE
CASCADE ON DELETE NO ACTION
);
```

---

5) Creating “ticket_booking_header” table:

```
CREATE TABLE ticket_booking_header(
     booking_id VARCHAR(20) NOT NULL,
     concert_id VARCHAR(20) NOT NULL,
     cust_id VARCHAR(20) NOT NULL,
     total_amount FLOAT DEFAULT 0,
     discount FLOAT DEFAULT 0,
     discount_amount FLOAT DEFAULT 0,
     tax FLOAT DEFAULT 0,
     tax_amount FLOAT DEFAULT 0,
     net_amount FLOAT DEFAULT 0,
     CHECK (discount>=0 AND tax>=0 AND discount_amount>=0 AND
tax_amount>=0),
     CHECK (total_amount>=0 AND net_amount>=0),
     PRIMARY KEY (booking_id),
     FOREIGN KEY (concert_id) REFERENCES concert(concert_id) ON UPDATE
CASCADE ON DELETE NO ACTION,
     FOREIGN KEY (cust_id) REFERENCES customer(cust_id) ON UPDATE CASCADE
ON DELETE NO ACTION
);
```

---

6) Creating “ticket_booking_details” table:

```
CREATE TABLE ticket_booking_details(
     seq_no INT NOT NULL AUTO_INCREMENT,
     booking_id VARCHAR(20) NOT NULL,
     ticketType_id VARCHAR(30) NOT NULL,
```

```
    tickets_bought INT,
    price FLOAT DEFAULT 0,
    amount FLOAT DEFAULT 0,
    CHECK (price>=0 AND amount>=0 AND tickets_bought>0),
    PRIMARY KEY (seq_no, booking_id),
    FOREIGN KEY (booking_id) REFERENCES ticket_booking_header(booking_id)
ON UPDATE CASCADE ON DELETE NO ACTION,
    FOREIGN KEY (ticketType_id) REFERENCES ticket_type(ticketType_id) ON
UPDATE CASCADE ON DELETE NO ACTION
);
```

INSERT COMMANDS

1) **Inserting values into "customer" table:**

```
INSERT INTO customer (cust_id, cust_name, email, phone_no, street_name, city, DOB, age)
VALUES ('CUST1', 'Richa Agarwal', 'richaa@abc.com', '9218756349', 'Eastwood Road',
'Bangalore', '2001-10-19', '21');
INSERT INTO customer (cust_id, cust_name, email, phone_no, street_name, city, DOB, age)
VALUES ('CUST2', 'Ankita Rajan', 'ankitar@abc.com',  '9400275418', 'Milan Road',
'Mumbai', '1995-11-10', '28');
INSERT INTO customer (cust_id, cust_name, email, phone_no, street_name, city, DOB, age)
VALUES ('CUST3', 'Sunita Raval', 'sunitaraval@xyz.com', '9491230333', 'Riddhipur Road',
'Guwahati', '2006-02-29','17');
INSERT INTO customer (cust_id, cust_name, email, phone_no, street_name, city, DOB, age)
VALUES ('CUST4', 'Naina Prabhu', 'nainaprabhu@rst.com', '8103065876', 'Vikhroli Road',
'Bhubaneshwar', '2000-09-01', '23');
INSERT INTO customer (cust_id, cust_name, email, phone_no, street_name, city, DOB, age)
VALUES ('CUST5', 'John Issac', 'johnissac@def.com', '8723197324', 'Doe Street',
'Kolkata', '1995-06-06', '28');
INSERT INTO customer (cust_id, cust_name, email, phone_no, street_name, city, DOB, age)
VALUES ('CUST6', 'Ayushi Savant', 's.ayushi@lmn.com', '9821780772', 'Dhirajgarh Marg',
'New Delhi', '2004-03-13', '19');
INSERT INTO customer (cust_id, cust_name, email, phone_no, street_name, city, DOB, age)
VALUES ('CUST7', 'Sakshi Dixit', 'sakshid@def.com', '8743107654', 'SV Road', 'Mysore',
'1996-01-31','27');
INSERT INTO customer (cust_id, cust_name, email, phone_no, street_name, city, DOB, age)
VALUES ('CUST8', 'Akhila Mohan', 'akhilamohan@rst.com', '8451850673', 'Aasth Chowk',
'Lucknow', '2008-04-15','15');
INSERT INTO customer (cust_id, cust_name, email, phone_no, street_name, city, DOB, age)
VALUES ('CUST9', 'Mona Shankar', 'mona.s@lmn.com', '6706272897', 'Sahil Nagar',
'Bangalore', '1998-08-04','25');
INSERT INTO customer (cust_id, cust_name, email, phone_no, street_name, city, DOB, age)
VALUES ('CUST10', 'Nandini Roy', 'nandiniroy@xyz.com', '7294654110', 'Doe Street',
'Chennai', '2007-05-25','16');
```

---

2) **Inserting values into "venues" table:**

```
INSERT INTO venues (venue_id, venue_name, capacity, street_name, city) VALUES ('V1',
'Phoenix Market City', '5000', 'Whitefield', 'Bangalore');
INSERT INTO venues (venue_id, venue_name, capacity, street_name, city) VALUES ('V2',
'Mahalaxmi Race Course', '6000', 'Keshavrao Khadye Marg', 'Mumbai');
INSERT INTO venues (venue_id, venue_name, capacity, street_name, city) VALUES ('V3',
'Vagator Beach', '20000', 'Bardez', 'Goa');
INSERT INTO venues (venue_id, venue_name, capacity, street_name, city) VALUES ('V4',
'Boomrang BarXsocial', '35000', 'Koramangala', 'Bangalore');
INSERT INTO venues (venue_id, venue_name, capacity, street_name, city) VALUES ('V5',
'Bhartiya Mall', '10000', 'Thanisandra Main Road', 'Bangalore');
```

```
INSERT INTO venues (venue_id, venue_name, capacity, street_name, city) VALUES ('V6',
'Nexus Shantiniketan', '15000', 'Whitefield', 'Bangalore');
INSERT INTO venues (venue_id, venue_name, capacity, street_name, city) VALUES ('V7',
'Inner Circle Ground', '15000', 'Whitefield', 'Bangalore');
INSERT INTO venues (venue_id, venue_name, capacity, street_name, city) VALUES ('V8', 'R2
Grounds', '3000', 'Bandra East', 'Mumbai');
INSERT INTO venues (venue_id, venue_name, capacity, street_name, city) VALUES ('V9',
'T.M.C Ground', '18000', 'Hiranandani Estate,Thane West', 'Thane');
INSERT INTO venues (venue_id, venue_name, capacity, street_name, city) VALUES ('V10',
'SVP Stadium', '160000', 'Worli', 'Mumbai');
```

---

3) Inserting values into "concert" table:

```
INSERT INTO concert (concert_id, venue_id, concert_name, date_of_concert, start_time,
end_time) VALUES ('C1', 'V7', 'Papon Live in Concert', '2024-11-25', '18:00:00',
'22:00:00');
INSERT INTO concert (concert_id, venue_id, concert_name, date_of_concert, start_time,
end_time) VALUES ('C2', 'V8', 'Ed Sheeran: +-=/* Tour', '2024-03-16', '18:30:00',
'23:00:00');
INSERT INTO concert (concert_id, venue_id, concert_name, date_of_concert, start_time,
end_time) VALUES ('C3', 'V1', 'Sunburn', '2024-12-28', '18:30:00', '23:00:00');
INSERT INTO concert (concert_id, venue_id, concert_name, date_of_concert, start_time,
end_time) VALUES ('C4', 'V1', 'Super Sunday Bollywood Night', '2024-11-19', '17:00:00',
'21:00:00');
INSERT INTO concert (concert_id, venue_id, concert_name, date_of_concert, start_time,
end_time) VALUES ('C5', 'V2', 'Ronan Keating', '2024-01-18', '19:00:00', '23:00:00');
INSERT INTO concert (concert_id, venue_id, concert_name, date_of_concert, start_time,
end_time) VALUES ('C6', 'V7', 'Whitefield Music Festival', '2024-02-25', '18:00:00',
'22:00:00');
INSERT INTO concert (concert_id, venue_id, concert_name, date_of_concert, start_time,
end_time) VALUES ('C7', 'V9', 'Neeti Mohan', '2024-12-16', '17:00:00', '21:00:00');
INSERT INTO concert (concert_id, venue_id, concert_name, date_of_concert, start_time,
end_time) VALUES ('C8', 'V5', 'Sunburn Arena Ft. Dimitri Vegas & Like Mike-Mumbai',
'2024-11-24', '18:30:00', '23:00:00');
INSERT INTO concert (concert_id, venue_id, concert_name, date_of_concert, start_time,
end_time) VALUES ('C9', 'V5', 'Sonu Nigam', '2024-12-02', '17:30:00', '20:30:00');
INSERT INTO concert (concert_id, venue_id, concert_name, date_of_concert, start_time,
end_time) VALUES ('C10', 'V6', 'KING', '2024-12-24', '19:30:00', '23:30:00');
```

---

4) Inserting values into "ticket_type" table:

```
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T1', 'C1', '1000', '6000');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T2', 'C1', '1800', '5200');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T3', 'C1', '2500', '3000');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T4', 'C1', '4000', '800');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T1', 'C2', '800', '3000');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T1', 'C3', '500', '3000');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T2', 'C3', '1000', '1500');
```

```
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T3', 'C3', '1500', '500');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T1', 'C4', '800', '2500');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T2', 'C4', '1200', '2000');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T1', 'C5', '400', '400');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T2', 'C5', '1200', '3600');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T3', 'C5', '2000', '2000');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T1', 'C7', '1000', '7000');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T2', 'C7', '2800', '6000');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T3', 'C7', '3500', '3200');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T4', 'C7', '5000', '1800');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T1', 'C8', '1500', '5000');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T2', 'C8', '3200', '3200');
INSERT INTO ticket_type (ticketType_id, concert_id, price, tickets_available) VALUES
('T3', 'C8', '5000', '1800');
```

---

5) Inserting values into "ticket_booking_header" table:

```
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B1', 'C3', 'CUST6', 10, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B2', 'C2', 'CUST3', 20, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B3', 'C5', 'CUST8', 15, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B4', 'C1', 'CUST5', 5, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B5', 'C7', 'CUST1', 20, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B6', 'C3', 'CUST10', 15, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B7', 'C1', 'CUST2', 10, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B8', 'C5', 'CUST9', 5, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B9', 'C1', 'CUST4', 10, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B10', 'C8', 'CUST7', 15, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B11', 'C4', 'CUST1', 20, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B12', 'C8', 'CUST2', 5, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B13', 'C2', 'CUST3', 15, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B14', 'C7', 'CUST4', 20, 5);
```

```
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B15', 'C4', 'CUST6', 10, 5);
INSERT INTO ticket_booking_header (booking_id, concert_id, cust_id, discount, tax) VALUES
('B16', 'C7', 'CUST8', 5, 5);
```

---

6) Inserting values into "ticket_booking_details" table:

```
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B1', 'T2', 2);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B1', 'T3', 1);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B2', 'T1', 4);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B3', 'T1', 3);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B3', 'T2', 2);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B4', 'T4', 1);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B4', 'T3', 2);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B4', 'T2', 3);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B5', 'T1', 3);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B5', 'T2', 2);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B6', 'T1', 6);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B7', 'T3', 2);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B7', 'T4', 1);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B7', 'T1', 4);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B7', 'T2', 3);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B8', 'T3', 3);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B9', 'T3', 3);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B9', 'T1', 5);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B10', 'T1', 3);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B10', 'T2', 2);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B10', 'T3', 1);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B11', 'T2', 4);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B12', 'T3', 3);
```

```
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B13', 'T1', 5);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B14', 'T2', 1);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B14', 'T1', 6);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B14', 'T3', 1);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B15', 'T2', 4);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B16', 'T3', 2);
INSERT INTO ticket_booking_details (booking_id, ticketType_id, tickets_bought) VALUES
('B16', 'T4', 3);
```

CRUD OPERATIONS

1) Inserting a value into “customer” table:

```
mysql> INSERT INTO customer VALUES ('CUST11', 'Anuha Patil', 'anuha@gmail.com', '123456789', 'B Street', 'Bangalore', '2002-11-12', 22);
Query OK, 1 row affected (0.01 sec)
```

```
mysql> SELECT * FROM customer;
+---------+---------------+--------------------+------------+-----------------+-------------+------------+------+
| cust_id | cust_name     | email              | phone_no   | street_name     | city        | DOB        | age  |
+---------+---------------+--------------------+------------+-----------------+-------------+------------+------+
| CUST1   | Richa Agarwal | richaa@abc.com     | 9218756349 | Eastwood Road   | Bangalore   | 2001-10-19 |   21 |
| CUST10  | Nandini Roy   | nandiniroy@xyz.com | 7294654110 | Doe Street      | Chennai     | 2007-05-25 |   16 |
| CUST11  | Anuha Patil   | anuha@gmail.com    | 123456789  | B Street        | Bangalore   | 2002-11-12 |   22 |
| CUST2   | Ankita Rajan  | ankitar@abc.com    | 9400275418 | Milan Road      | Mumbai      | 1995-11-10 |   28 |
| CUST4   | Naina Prabhu  | nainaprabhu@rst.com| 8103065876 | Vikhroli Road   | Bhubaneshwar| 2000-09-01 |   23 |
| CUST5   | John Issac    | johnissac@def.com  | 8723197324 | Doe Street      | Kolkata     | 1995-06-06 |   28 |
| CUST6   | Ayushi Savant | s.ayushi@lmn.com   | 9821780772 | Dhirajgarh Marg | New Delhi   | 2004-03-13 |   19 |
| CUST7   | Sakshi Dixit  | sakshid@def.com    | 8743107654 | SV Road         | Mysore      | 1996-01-31 |   27 |
| CUST8   | Akhila Mohan  | akhilamohan@rst.com| 8451850673 | Aasth Chowk     | Lucknow     | 2008-04-15 |   15 |
| CUST9   | Mona Shankar  | mona.s@lmn.com     | 6706272897 | Sahil Nagar     | Bangalore   | 1998-08-04 |   25 |
+---------+---------------+--------------------+------------+-----------------+-------------+------------+------+
10 rows in set (0.00 sec)
```

2) Reading all concerts available in the database:

```
mysql> SELECT * from concert;
+------------+----------+-------------------------------+-----------------+------------+----------+
| concert_id | venue_id | concert_name                  | date_of_concert | start_time | end_time |
+------------+----------+-------------------------------+-----------------+------------+----------+
| C1         | V7       | Papon Live in Concert         | 2024-11-25      | 18:00:00   | 22:00:00 |
| C10        | V6       | KING                          | 2024-12-24      | 19:30:00   | 23:30:00 |
| C11        | V1       | Aatmatrisha                   | 2023-05-12      | 18:00:00   | 22:00:00 |
| C2         | V8       | Ed Sheeran: +-=/* Tour        | 2024-03-16      | 18:30:00   | 23:00:00 |
| C3         | V1       | Sunburn                       | 2024-12-28      | 18:30:00   | 23:00:00 |
| C4         | V1       | Super Sunday Bollywood Night  | 2024-11-19      | 17:00:00   | 21:00:00 |
| C5         | V2       | Ronan Keating                 | 2024-01-18      | 19:00:00   | 23:00:00 |
| C6         | V7       | Whitefield Music Festival     | 2024-02-25      | 18:00:00   | 22:00:00 |
| C7         | V9       | Neeti Mohan                   | 2024-12-16      | 17:00:00   | 21:00:00 |
| C9         | V5       | Sonu Nigam                    | 2024-12-02      | 17:30:00   | 20:30:00 |
+------------+----------+-------------------------------+-----------------+------------+----------+
10 rows in set (0.00 sec)
```

3) Updating capacity of venue V2 to 6000:

```
mysql> UPDATE venues
    -> SET capacity = '6000'
    -> WHERE venue_id = 'V2';
Query OK, 0 rows affected (0.00 sec)
Rows matched: 1  Changed: 0  Warnings: 0

mysql> SELECT * FROM venues;
+----------+----------------------+----------+-----------+------------------------------+
| venue_id | venue_name           | capacity | city      | street_name                  |
+----------+----------------------+----------+-----------+------------------------------+
| V1       | Phoenix Market City  |     5000 | Bangalore | Whitefield                   |
| V10      | SVP Stadium          |   160000 | Mumbai    | Worli                        |
| V2       | Mahalaxmi Race Course|     6000 | Mumbai    | Keshavrao Khadye Marg        |
| V3       | Vagator Beach        |    20000 | Goa       | Bardez                       |
| V4       | Boomrang BarXsocial  |    35000 | Bangalore | Koramangala                  |
| V5       | Bhartiya Mall        |    10000 | Bangalore | Thanisandra Main Road        |
| V6       | Nexus Shantiniketan  |    15000 | Bangalore | Whitefield                   |
| V7       | Inner Circle Ground  |    15000 | Bangalore | Whitefield                   |
| V8       | R2 Grounds           |     3000 | Mumbai    | Bandra East                  |
| V9       | T.M.C Ground         |    18000 | Thane     | Hiranandani Estate,Thane West |
+----------+----------------------+----------+-----------+------------------------------+
10 rows in set (0.00 sec)
```

4) Deleting ticket_type for a concert:

```
mysql> DELETE FROM ticket_type WHERE ticketType_id = 'T5' AND concert_id = 'C11';
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT * FROM ticket_type;
+---------------+------------+-------+-------------------+
| ticketType_id | concert_id | price | tickets_available |
+---------------+------------+-------+-------------------+
| T1            | C1         |  1000 |              6000 |
| T1            | C2         |   800 |              3000 |
| T1            | C3         |   500 |              3000 |
| T1            | C4         |   800 |              2500 |
| T1            | C5         |   400 |               400 |
| T1            | C7         |  1000 |              7000 |
| T2            | C1         |  1800 |              5200 |
| T2            | C3         |  1000 |              1500 |
| T2            | C4         |  1200 |              2000 |
| T2            | C5         |  1200 |              3600 |
| T2            | C7         |  2800 |              6000 |
| T3            | C1         |  2500 |              3000 |
| T3            | C3         |  1500 |               500 |
| T3            | C5         |  2000 |              2000 |
| T3            | C7         |  3500 |              3200 |
| T4            | C1         |  4000 |               800 |
| T4            | C7         |  5000 |              1800 |
+---------------+------------+-------+-------------------+
17 rows in set (0.00 sec)
```

JOIN OPERATIONS

1) Inner Join - List all concerts with their venues:

```
mysql> SELECT concert.concert_id, concert.concert_name, concert.date_of_concert, venues.venue_name
    ->     FROM concert
    ->     INNER JOIN venues ON concert.venue_id = venues.venue_id;
+------------+------------------------------+-----------------+----------------------+
| concert_id | concert_name                 | date_of_concert | venue_name           |
+------------+------------------------------+-----------------+----------------------+
| C1         | Papon Live in Concert        | 2024-11-25      | Inner Circle Ground  |
| C10        | KING                         | 2024-12-24      | Nexus Shantiniketan  |
| C11        | Aatmatrisha                  | 2023-05-12      | Phoenix Market City  |
| C2         | Ed Sheeran: +-=/* Tour       | 2024-03-16      | R2 Grounds           |
| C3         | Sunburn                      | 2024-12-28      | Phoenix Market City  |
| C4         | Super Sunday Bollywood Night | 2024-11-19      | Phoenix Market City  |
| C5         | Ronan Keating                | 2024-01-18      | Mahalaxmi Race Course |
| C6         | Whitefield Music Festival    | 2024-02-25      | Inner Circle Ground  |
| C7         | Neeti Mohan                  | 2024-12-16      | T.M.C Ground         |
| C9         | Sonu Nigam                   | 2024-12-02      | Bhartiya Mall        |
+------------+------------------------------+-----------------+----------------------+
10 rows in set (0.00 sec)
```

2) Left Join - List all concerts and their bookings, even if there are no bookings:

```
mysql> SELECT concert.concert_id, concert.concert_name, concert.date_of_concert, ticket_booking_header.booking_id
    ->     FROM concert
    ->     LEFT JOIN ticket_booking_header ON concert.concert_id = ticket_booking_header.concert_id;
+------------+------------------------------+-----------------+------------+
| concert_id | concert_name                 | date_of_concert | booking_id |
+------------+------------------------------+-----------------+------------+
| C1         | Papon Live in Concert        | 2024-11-25      | B4         |
| C1         | Papon Live in Concert        | 2024-11-25      | B7         |
| C1         | Papon Live in Concert        | 2024-11-25      | B9         |
| C10        | KING                         | 2024-12-24      | NULL       |
| C11        | Aatmatrisha                  | 2023-05-12      | NULL       |
| C2         | Ed Sheeran: +-=/* Tour       | 2024-03-16      | NULL       |
| C3         | Sunburn                      | 2024-12-28      | B1         |
| C3         | Sunburn                      | 2024-12-28      | B6         |
| C4         | Super Sunday Bollywood Night | 2024-11-19      | B11        |
| C4         | Super Sunday Bollywood Night | 2024-11-19      | B15        |
| C5         | Ronan Keating                | 2024-01-18      | B3         |
| C5         | Ronan Keating                | 2024-01-18      | B8         |
| C6         | Whitefield Music Festival    | 2024-02-25      | NULL       |
| C7         | Neeti Mohan                  | 2024-12-16      | B14        |
```

3) Right Join - List all venues and concerts taking place at the venues:

```
mysql> SELECT venues.venue_id, venues.venue_name, concert.concert_id, concert.concert_name, concert.date_of_concert
    ->     FROM venues
    ->     RIGHT JOIN concert ON venues.venue_id = concert.venue_id;
+----------+----------------------+------------+------------------------------+-----------------+
| venue_id | venue_name           | concert_id | concert_name                 | date_of_concert |
+----------+----------------------+------------+------------------------------+-----------------+
| V7       | Inner Circle Ground  | C1         | Papon Live in Concert        | 2024-11-25      |
| V6       | Nexus Shantiniketan  | C10        | KING                         | 2024-12-24      |
| V1       | Phoenix Market City  | C11        | Aatmatrisha                  | 2023-05-12      |
| V8       | R2 Grounds           | C2         | Ed Sheeran: +-=/* Tour       | 2024-03-16      |
| V1       | Phoenix Market City  | C3         | Sunburn                      | 2024-12-28      |
| V1       | Phoenix Market City  | C4         | Super Sunday Bollywood Night | 2024-11-19      |
| V2       | Mahalaxmi Race Course | C5         | Ronan Keating                | 2024-01-18      |
| V7       | Inner Circle Ground  | C6         | Whitefield Music Festival    | 2024-02-25      |
| V9       | T.M.C Ground         | C7         | Neeti Mohan                  | 2024-12-16      |
| V5       | Bhartiya Mall        | C9         | Sonu Nigam                   | 2024-12-02      |
+----------+----------------------+------------+------------------------------+-----------------+
10 rows in set (0.00 sec)
```

4) Self Join - Find concerts in the same venue on the same date:

```
mysql>  SELECT a.concert_id AS concert1, a.concert_name AS name1, b.concert_id AS concert2, b.concert_name AS name2
    ->    FROM concert a, concert b
    ->    WHERE a.venue_id = b.venue_id AND a.date_of_concert = b.date_of_concert AND a.concert_id <> b.concert_id;
Empty set (0.00 sec)
```

5) Cross Join - List all possible combinations of concerts and venues:

```
mysql>   SELECT concert.concert_id, concert.concert_name, venues.venue_id, venues.venue_name
    ->    FROM concert
    ->    CROSS JOIN venues;
+------------+------------------------------+----------+----------------------+
| concert_id | concert_name                 | venue_id | venue_name           |
+------------+------------------------------+----------+----------------------+
| C9         | Sonu Nigam                   | V1       | Phoenix Market City  |
| C7         | Neeti Mohan                  | V1       | Phoenix Market City  |
| C6         | Whitefield Music Festival    | V1       | Phoenix Market City  |
| C5         | Ronan Keating                | V1       | Phoenix Market City  |
| C4         | Super Sunday Bollywood Night | V1       | Phoenix Market City  |
| C3         | Sunburn                      | V1       | Phoenix Market City  |
| C2         | Ed Sheeran: +-=/* Tour       | V1       | Phoenix Market City  |
| C11        | Aatmatrisha                  | V1       | Phoenix Market City  |
| C10        | KING                         | V1       | Phoenix Market City  |
| C1         | Papon Live in Concert        | V1       | Phoenix Market City  |
| C9         | Sonu Nigam                   | V10      | SVP Stadium          |
| C7         | Neeti Mohan                  | V10      | SVP Stadium          |
| C6         | Whitefield Music Festival    | V10      | SVP Stadium          |
| C5         | Ronan Keating                | V10      | SVP Stadium          |
| C4         | Super Sunday Bollywood Night | V10      | SVP Stadium          |
| C3         | Sunburn                      | V10      | SVP Stadium          |
| C2         | Ed Sheeran: +-=/* Tour       | V10      | SVP Stadium          |
| C11        | Aatmatrisha                  | V10      | SVP Stadium          |
| C10        | KING                         | V10      | SVP Stadium          |
| C1         | Papon Live in Concert        | V10      | SVP Stadium          |
| C9         | Sonu Nigam                   | V2       | Mahalaxmi Race Course |
| C7         | Neeti Mohan                  | V2       | Mahalaxmi Race Course |
| C6         | Whitefield Music Festival    | V2       | Mahalaxmi Race Course |
| C5         | Ronan Keating                | V2       | Mahalaxmi Race Course |
| C4         | Super Sunday Bollywood Night | V2       | Mahalaxmi Race Course |
| C3         | Sunburn                      | V2       | Mahalaxmi Race Course |
| C2         | Ed Sheeran: +-=/* Tour       | V2       | Mahalaxmi Race Course |
| C11        | Aatmatrisha                  | V2       | Mahalaxmi Race Course |
| C10        | KING                         | V2       | Mahalaxmi Race Course |
| C1         | Papon Live in Concert        | V2       | Mahalaxmi Race Course |
| C9         | Sonu Nigam                   | V3       | Vagator Beach        |
| C7         | Neeti Mohan                  | V3       | Vagator Beach        |
| C6         | Whitefield Music Festival    | V3       | Vagator Beach        |
| C5         | Ronan Keating                | V3       | Vagator Beach        |
| C4         | Super Sunday Bollywood Night | V3       | Vagator Beach        |
| C3         | Sunburn                      | V3       | Vagator Beach        |
| C2         | Ed Sheeran: +-=/* Tour       | V3       | Vagator Beach        |
| C11        | Aatmatrisha                  | V3       | Vagator Beach        |
| C10        | KING                         | V3       | Vagator Beach        |
```

FUNCTIONALITY QUERIES

1) Calculating the revenue generated by concerts at a particular venue:

```
mysql> SELECT c.concert_id, c.concert_name, c.date_of_concert, FORMAT(SUM(tbh.net_amount), 2) AS 'revenue_generated'
    -> FROM concert c
    -> JOIN ticket_booking_header tbh ON c.concert_id = tbh.concert_id
    -> WHERE c.venue_id = (
    ->     SELECT venue_id FROM venues WHERE venue_name = 'Phoenix Market City'
    -> )
    -> GROUP BY c.concert_id
    -> ORDER BY SUM(tbh.net_amount) DESC;
+------------+-----------------------------+----------------+-------------------+
| concert_id | concert_name                | date_of_concert | revenue_generated |
+------------+-----------------------------+----------------+-------------------+
| C3         | Sunburn                     | 2024-12-28      | 0.00              |
| C4         | Super Sunday Bollywood Night | 2024-11-19      | 0.00              |
+------------+-----------------------------+----------------+-------------------+
2 rows in set (0.01 sec)
```

2) Of all the events, finding the concert with the highest generated revenue:

```
mysql> SELECT c.concert_id, c.concert_name, c.date_of_concert, FORMAT(SUM(tbh.net_amount), 2) AS 'max_revenue_generated'
    -> FROM concert c
    -> JOIN ticket_booking_header tbh ON c.concert_id = tbh.concert_id
    -> GROUP BY c.concert_id
    -> ORDER BY SUM(tbh.net_amount) DESC
    -> LIMIT 1;
+------------+---------------------+----------------+----------------------+
| concert_id | concert_name        | date_of_concert | max_revenue_generated |
+------------+---------------------+----------------+----------------------+
| C1         | Papon Live in Concert | 2024-11-25      | 1,995.00             |
+------------+---------------------+----------------+----------------------+
1 row in set (0.00 sec)
```

3) Listing all events where the sum of prices of ticket classes is 3000:

```
mysql> SELECT c.concert_id, c.concert_name, c.date_of_concert
    -> FROM concert c
    -> WHERE c.concert_id IN (
    ->     SELECT t.concert_id FROM ticket_type t
    ->     GROUP BY t.concert_id
    ->     HAVING SUM(t.price) = 3000
    -> );
+------------+--------------+-----------------+
| concert_id | concert_name | date_of_concert |
+------------+--------------+-----------------+
| C3         | Sunburn      | 2024-12-28      |
+------------+--------------+-----------------+
1 row in set (0.00 sec)
```

4) Finding all customers who have purchased more than 7 tickets:

```
mysql> SELECT tbh.cust_id, c.cust_name, SUM(tbd.tickets_bought) AS 'no_of_tickets'
    -> FROM ticket_booking_header tbh
    -> JOIN ticket_booking_details tbd ON tbh.booking_id = tbd.booking_id
    -> JOIN customer c ON tbh.cust_id = c.cust_id
    -> GROUP BY tbh.cust_id
    -> HAVING SUM(tbd.tickets_bought) > 7;
+---------+--------------+---------------+
| cust_id | cust_name    | no_of_tickets |
+---------+--------------+---------------+
| CUST1   | Richa Agarwal |             9 |
| CUST2   | Ankita Rajan  |            10 |
| CUST4   | Naina Prabhu  |            16 |
| CUST5   | John Issac    |            10 |
| CUST8   | Akhila Mohan  |            10 |
+---------+--------------+---------------+
5 rows in set (0.00 sec)
```

5) Finding all venues at which concerts are taking place in December:

```
mysql> SELECT v.venue_id, v.venue_name
    -> FROM venues v
    -> WHERE v.venue_id IN (
    ->     SELECT DISTINCT c.venue_id FROM concert c
    ->     WHERE (c.date_of_concert >= '2024-12-01' AND c.date_of_concert <= '2024-12-31')
    -> );
+----------+---------------------+
| venue_id | venue_name          |
+----------+---------------------+
| V6       | Nexus Shantiniketan |
| V1       | Phoenix Market City |
| V9       | T.M.C Ground        |
| V5       | Bhartiya Mall       |
+----------+---------------------+
4 rows in set (0.00 sec)
```

6) Listing all the venues and the concerts taking place at the venues:

```
mysql> SELECT v.venue_id, v.venue_name, c.concert_id, c.concert_name, c.date_of_concert
    -> FROM venues v
    -> LEFT JOIN concert c ON v.venue_id = c.venue_id;
+----------+----------------------+------------+-----------------------------+-----------------+
| venue_id | venue_name           | concert_id | concert_name                | date_of_concert |
+----------+----------------------+------------+-----------------------------+-----------------+
| V1       | Phoenix Market City  | C11        | Aatmatrisha                 | 2023-05-12      |
| V1       | Phoenix Market City  | C3         | Sunburn                     | 2024-12-28      |
| V1       | Phoenix Market City  | C4         | Super Sunday Bollywood Night | 2024-11-19      |
| V10      | SVP Stadium          | NULL       | NULL                        | NULL            |
| V2       | Mahalaxmi Race Course | C5         | Ronan Keating               | 2024-01-18      |
| V3       | Vagator Beach        | NULL       | NULL                        | NULL            |
| V4       | Boomrang BarXsocial  | NULL       | NULL                        | NULL            |
| V5       | Bhartiya Mall        | C9         | Sonu Nigam                  | 2024-12-02      |
| V6       | Nexus Shantiniketan  | C10        | KING                        | 2024-12-24      |
| V7       | Inner Circle Ground  | C1         | Papon Live in Concert       | 2024-11-25      |
| V7       | Inner Circle Ground  | C6         | Whitefield Music Festival   | 2024-02-25      |
| V8       | R2 Grounds           | C2         | Ed Sheeran: +-=/* Tour      | 2024-03-16      |
| V9       | T.M.C Ground         | C7         | Neeti Mohan                 | 2024-12-16      |
+----------+----------------------+------------+-----------------------------+-----------------+
13 rows in set (0.00 sec)
```

7) Find all customers who attended a particular concert at a particular event:

```
mysql>
mysql> SELECT DISTINCT c.cust_id, c.cust_name
    -> FROM customer c
    -> JOIN ticket_booking_header tbh ON c.cust_id = tbh.cust_id
    -> JOIN concert con ON tbh.concert_id = con.concert_id
    -> JOIN venues v ON con.venue_id = v.venue_id
    -> WHERE v.venue_name = 'Phoenix Market City';
+---------+---------------+
| cust_id | cust_name     |
+---------+---------------+
| CUST6   | Ayushi Savant |
| CUST10  | Nandini Roy   |
| CUST1   | Richa Agarwal |
+---------+---------------+
3 rows in set (0.00 sec)
```

PROCEDURES, TRIGGERS, FUNCTIONS

1) Procedures:

a) To update ticket availability:

```
DELIMITER //
CREATE PROCEDURE UpdateTicketAvailability(ticketTypeID VARCHAR(20), eventID VARCHAR(20), ticketsBought INT)
BEGIN
    UPDATE ticket_type
    SET tickets_available = tickets_available - ticketsBought
    WHERE ticketType_id = ticketTypeID AND event_id = eventID;
END //
DELIMITER ;
```

b) To cancel booking:

```
DELIMITER //
CREATE PROCEDURE CancelBooking(bookingID VARCHAR(20))
BEGIN
    DECLARE ticketTypeID VARCHAR(20);
    DECLARE ticketsBought INT;

    -- Get ticket type and number of tickets bought for the booking
    SELECT ticketType_id, tickets_bought
    INTO ticketTypeID, ticketsBought
    FROM ticket_booking_details
    WHERE booking_id = bookingID;

    -- Update ticket availability
    UPDATE ticket_type
    SET tickets_available = tickets_available + ticketsBought
    WHERE ticketType_id = ticketTypeID;

    -- Delete booking details
    DELETE FROM ticket_booking_details
    WHERE booking_id = bookingID;

    -- Delete booking header
    DELETE FROM ticket_booking_header
    WHERE booking_id = bookingID;
END //
DELIMITER ;
```

2) Functions:

a) To get concert revenue:

```
DELIMITER //
CREATE FUNCTION GetEventRevenue(concert_id VARCHAR(20))
RETURNS FLOAT
BEGIN
    DECLARE revenue FLOAT;

    SELECT SUM(net_amount)
    INTO revenue
    FROM ticket_booking_header
    WHERE concert_id = concert_id;

    RETURN revenue;
END //
DELIMITER ;
```

b) To get customer details:

```
DELIMITER //
CREATE FUNCTION GetCustomerDetails(customerID VARCHAR(20))
RETURNS VARCHAR(255)
BEGIN
    DECLARE customerInfo VARCHAR(255);

    SELECT CONCAT(cust_name, ', ', email, ', ', telephone, ', ', address_line1, ', ', city, ', ', state, ', ', pincode)
    INTO customerInfo
    FROM customer
    WHERE cust_id = customerID;

    RETURN customerInfo;
END //
DELIMITER ;
```

3) Triggers:

a) To check validity of date:

```
DELIMITER $$

CREATE TRIGGER check_concert_date
BEFORE INSERT ON concert
FOR EACH ROW
BEGIN
    IF NEW.date_of_concert <= CURRENT_DATE() THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Invalid concert date';
    END IF;
END //

DELIMITER ;
```

b) To get price after discount and taxes:

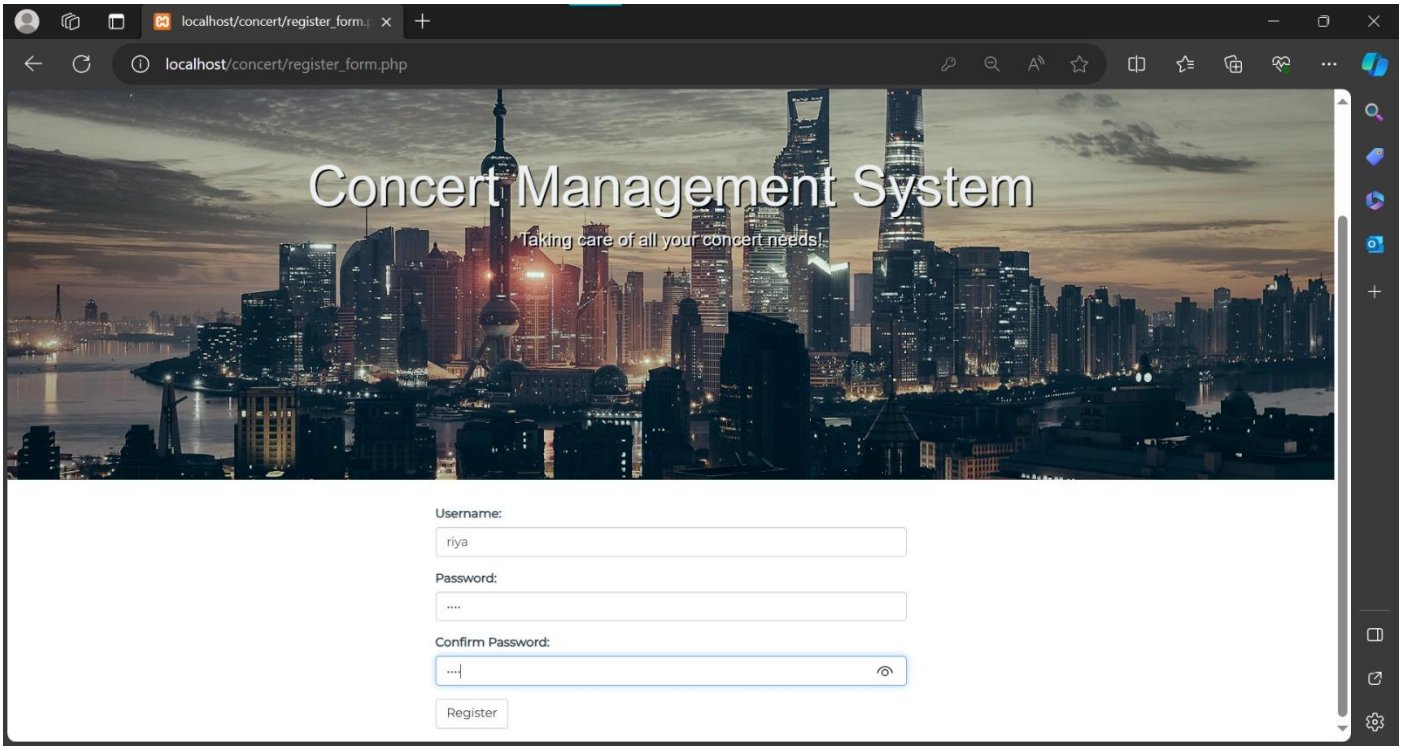
```
DELIMITER $$
CREATE TRIGGER ticket_amount
AFTER INSERT ON ticket_booking_details
FOR EACH ROW
BEGIN
    UPDATE ticket_booking_header AS a
    SET a.total_amount = a.total_amount + new.amount,
        a.discount_amount = a.total_amount * (a.discount/100),
        a.tax_amount = (a.total_amount - a.discount_amount) * (a.tax/100),
        a.net_amount = a.total_amount - a.discount_amount + a.tax_amount
    WHERE a.booking_id=new.booking_id;
END;$$
```

c) To set attributes price and amount:

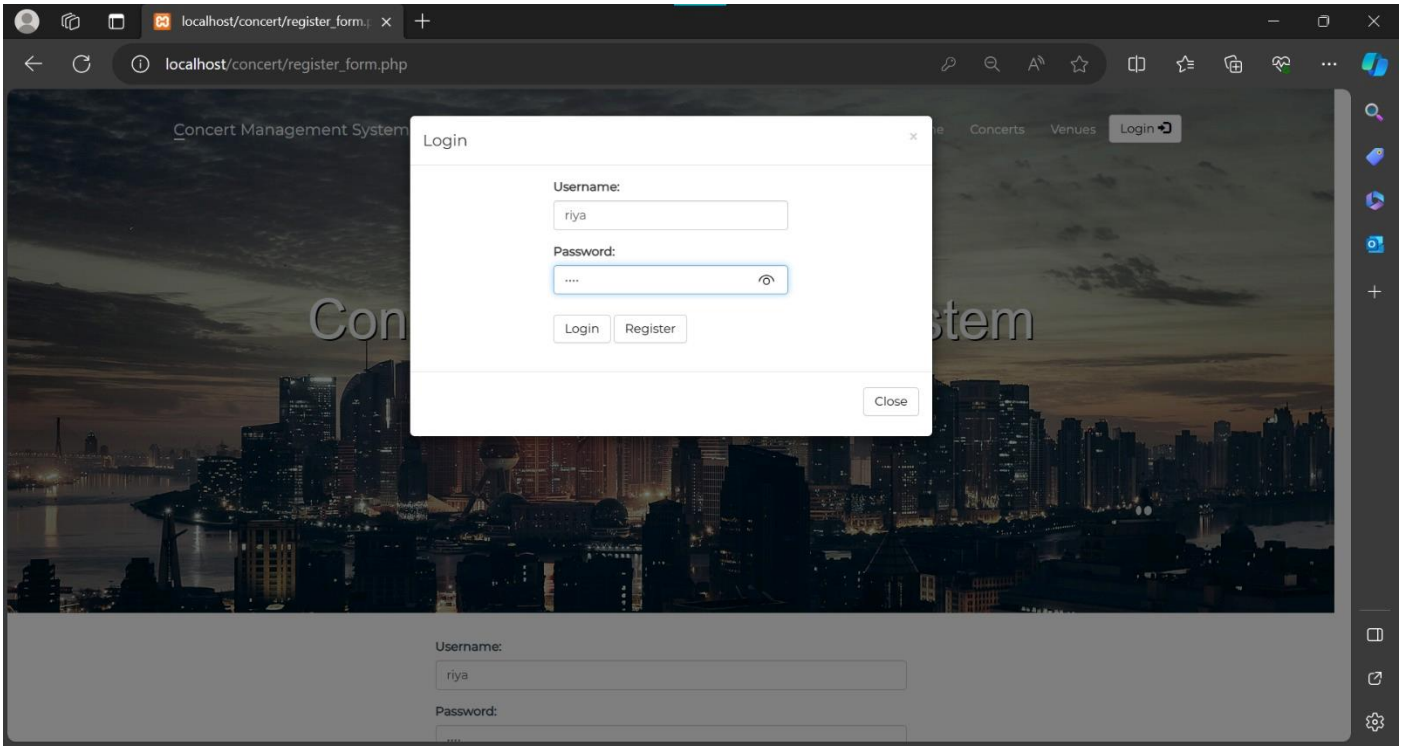
```
DELIMITER $$
CREATE TRIGGER get_price
BEFORE INSERT ON ticket_booking_details
FOR EACH ROW
BEGIN
    DECLARE V1 VARCHAR(20);
    DECLARE V2 INT;
    SELECT event_id INTO V1 FROM ticket_booking_header WHERE booking_id = new.booking_id;
    SELECT price INTO V2 FROM ticket_type WHERE ticketType_id = new.ticketType_id AND event_id = V1;
    SET new.price=V2;
    SET new.amount = new.price * new.tickets_bought;
END;$$
DELIMITER ;
```

GUI / FRONTEND

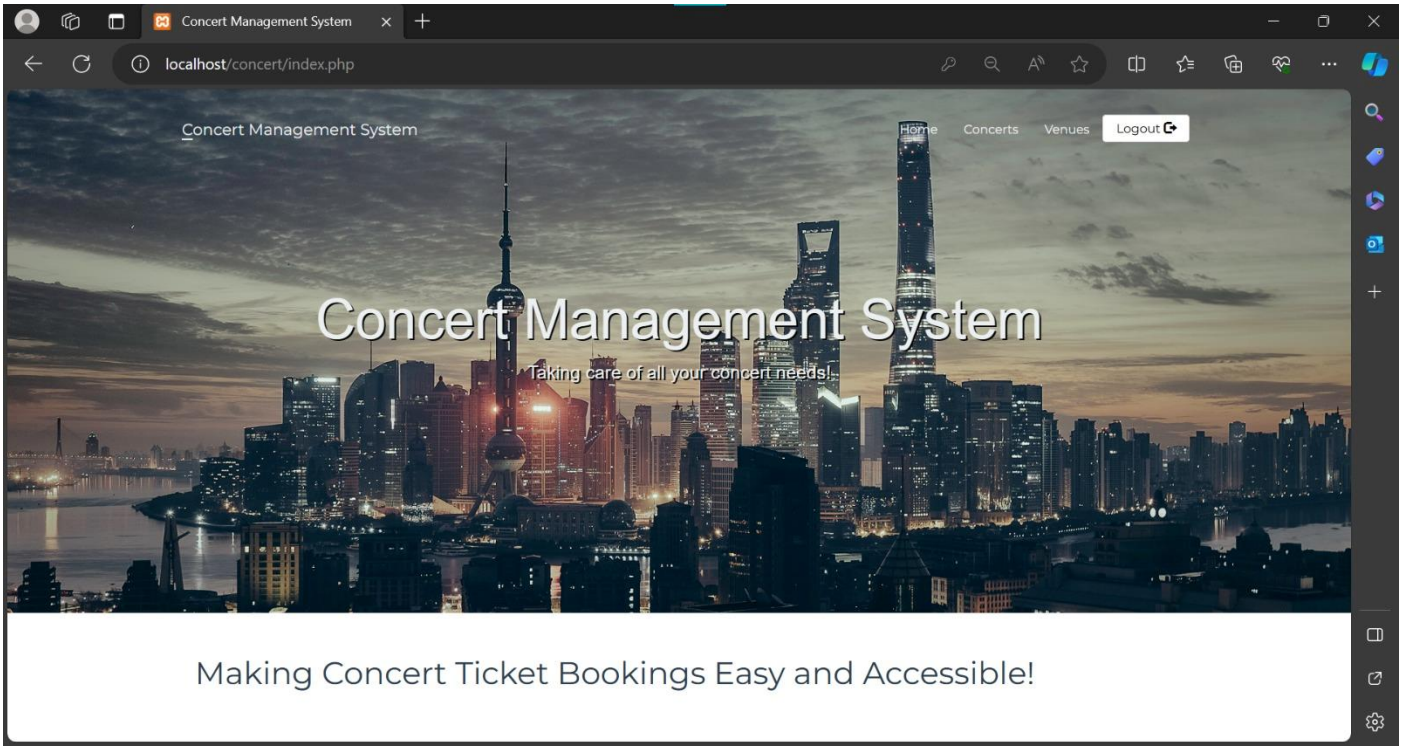
1) Register Page:



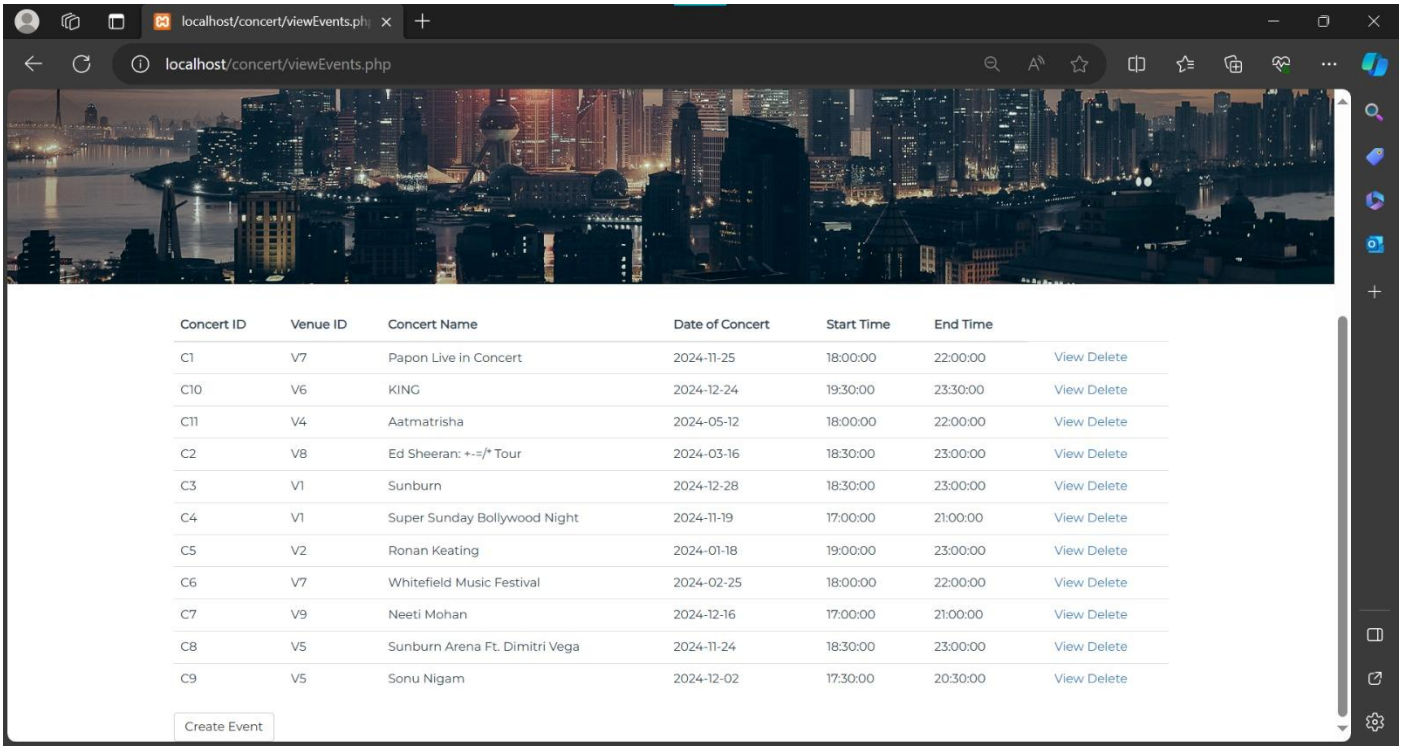
2) Login Page:



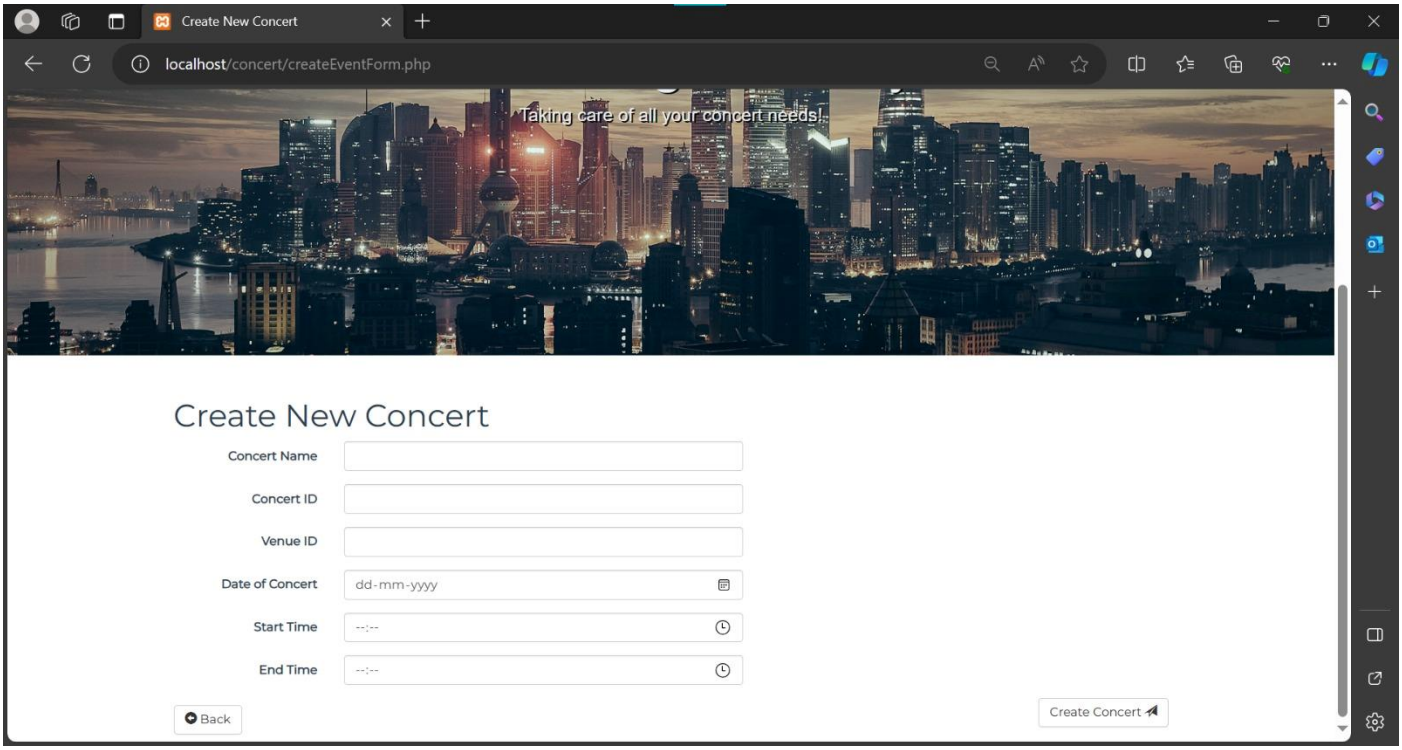
3) Home Page:



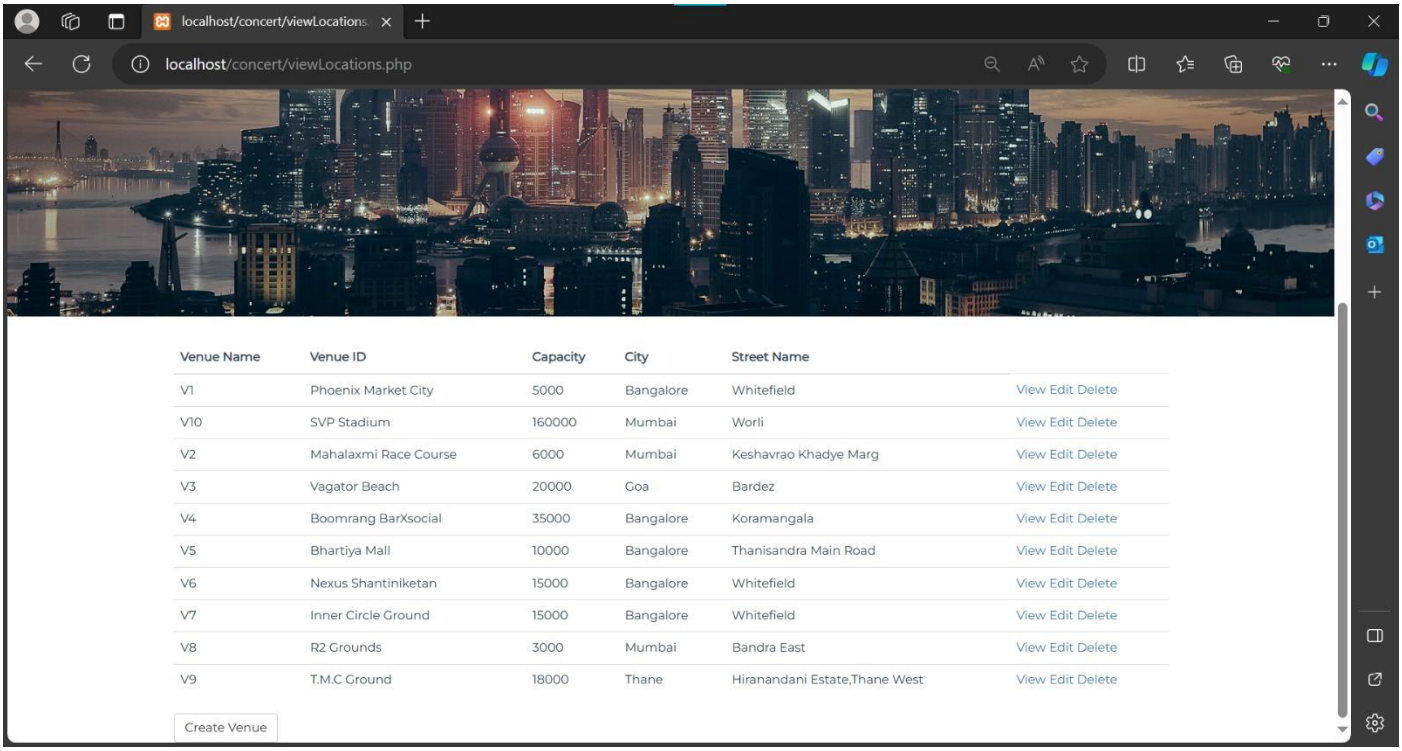
4) View Concerts Page:



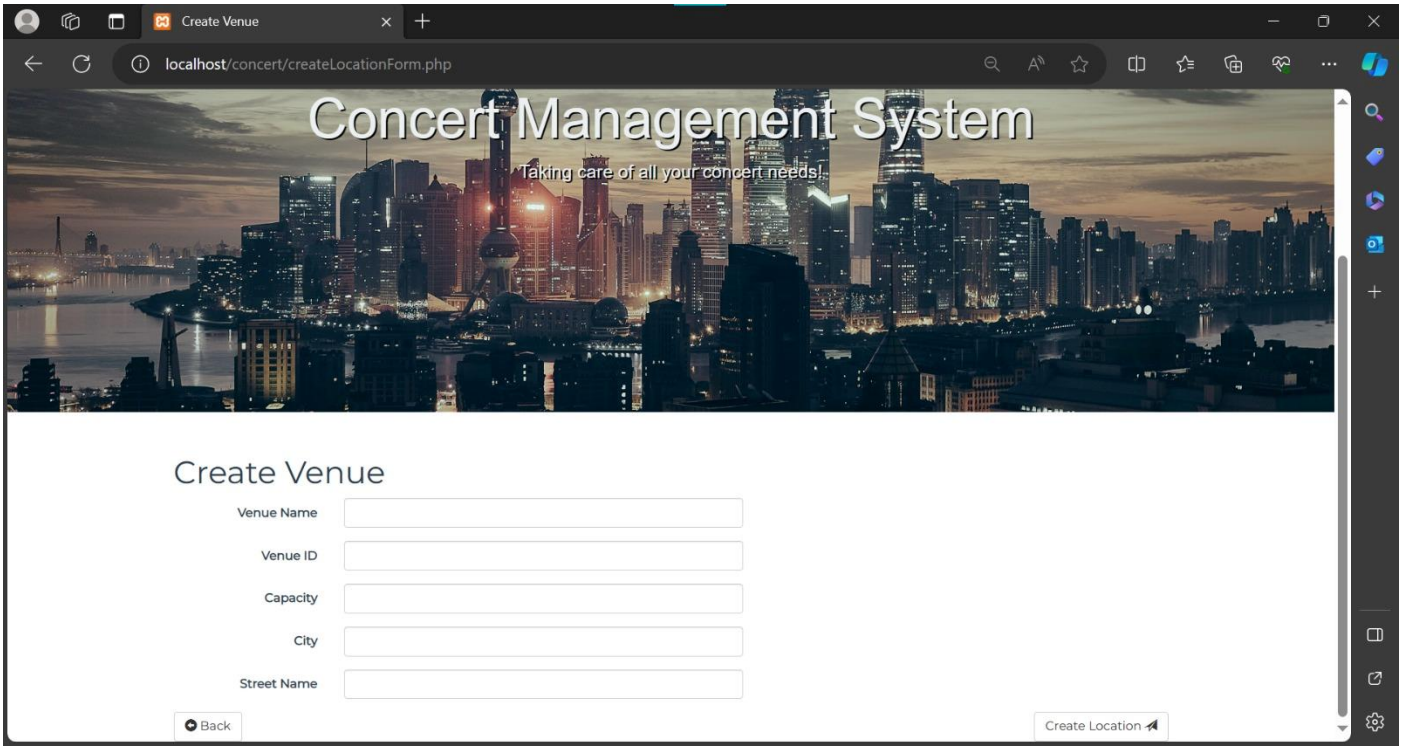
5) Create New Concert:



6) View Venues Page:



7) Create Venue Page:



*****