ASSIGNMENT TITLE:

Ticket Booking System : MySQL Assignment 5

SUBMITED BY:

BALAKUMARAN P

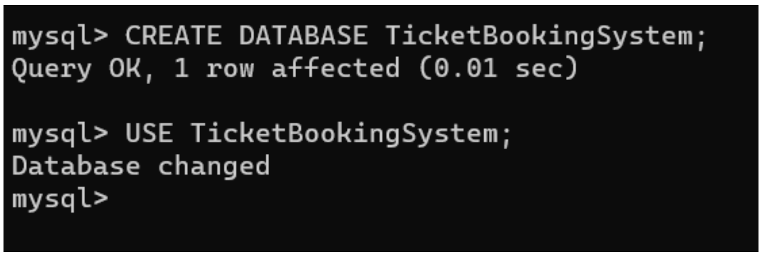
DATE OF SUBMISSION:

22/01/2024

<u>Ticket Booking System : MySQL Assignment 5</u>

Tasks 1: Database Design:

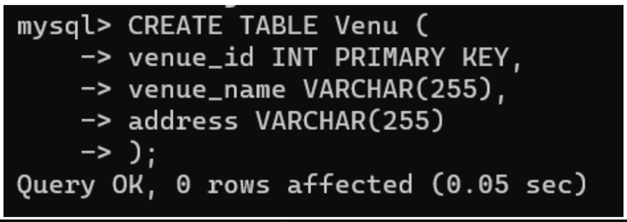
1. Create the database named "**TicketBookingSystem**"



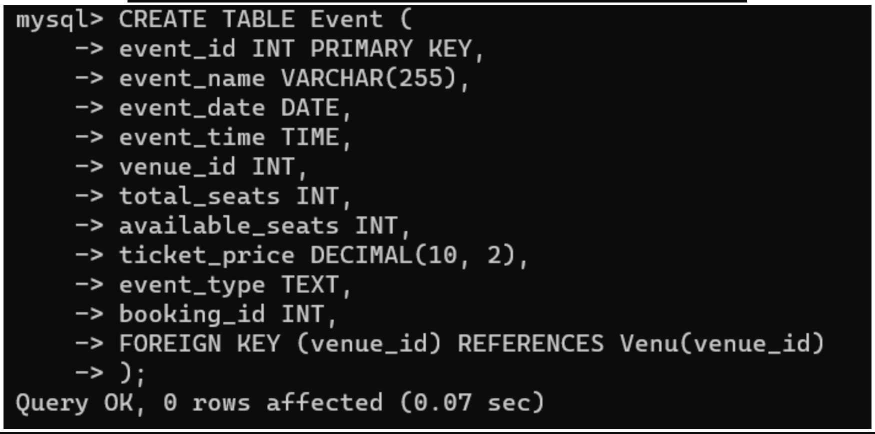
```
mysql> CREATE DATABASE TicketBookingSystem;
Query OK, 1 row affected (0.01 sec)

mysql> USE TicketBookingSystem;
Database changed
mysql>
```

2. Write SQL scripts to create the mentioned tables with appropriate data types, constraints, and relationships.

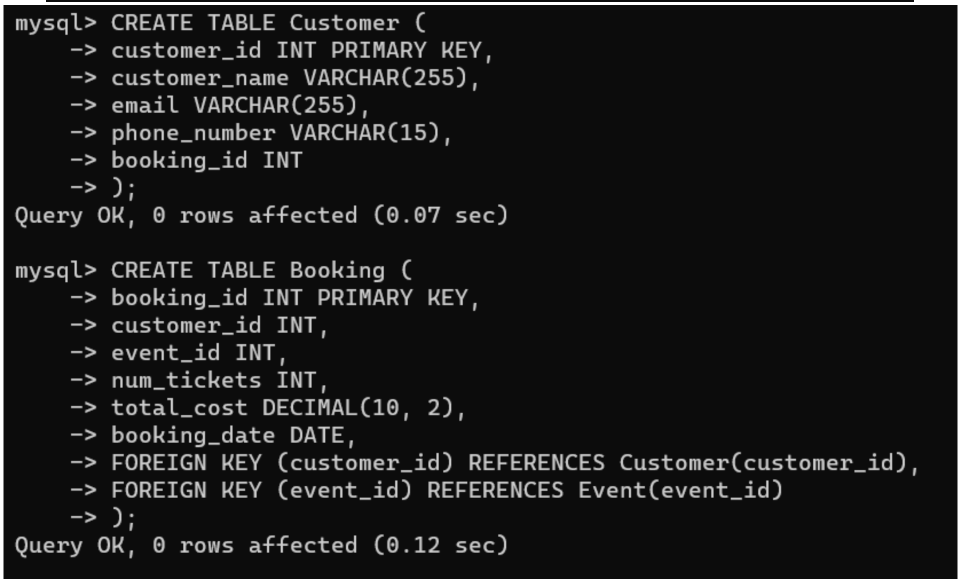
• Venu • Event • Customers • Booking



```
mysql> CREATE TABLE Venu (
    -> venue_id INT PRIMARY KEY,
    -> venue_name VARCHAR(255),
    -> address VARCHAR(255)
    -> );
Query OK, 0 rows affected (0.05 sec)
```

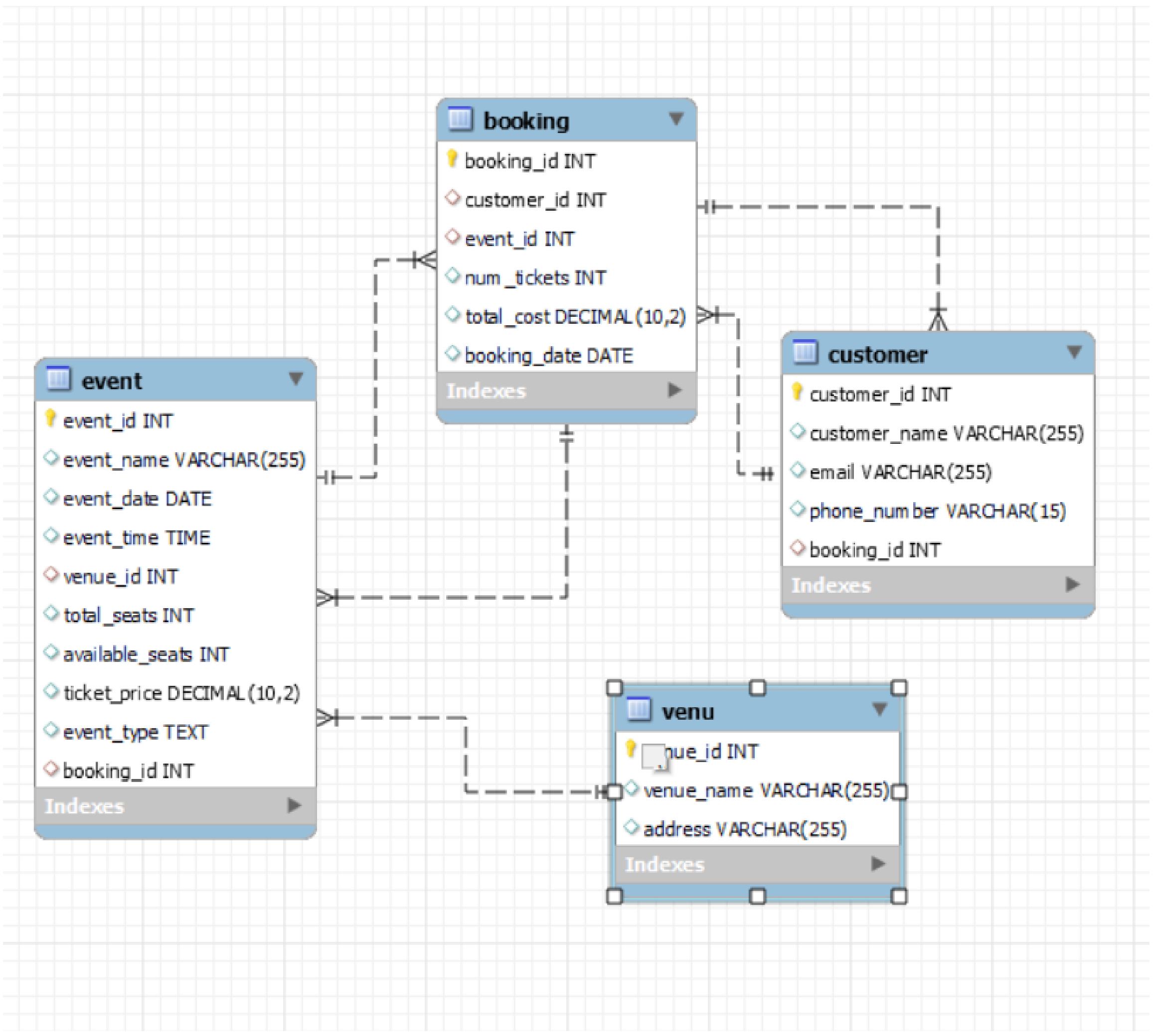


```
mysql> CREATE TABLE Event (
    -> event_id INT PRIMARY KEY,
    -> event_name VARCHAR(255),
    -> event_date DATE,
    -> event_time TIME,
    -> venue_id INT,
    -> total_seats INT,
    -> available_seats INT,
    -> ticket_price DECIMAL(10, 2),
    -> event_type TEXT,
    -> booking_id INT,
    -> FOREIGN KEY (venue_id) REFERENCES Venu(venue_id)
    -> );
Query OK, 0 rows affected (0.07 sec)
```



```
mysql> CREATE TABLE Customer (
    -> customer_id INT PRIMARY KEY,
    -> customer_name VARCHAR(255),
    -> email VARCHAR(255),
    -> phone_number VARCHAR(15),
    -> booking_id INT
    -> );
Query OK, 0 rows affected (0.07 sec)

mysql> CREATE TABLE Booking (
    -> booking_id INT PRIMARY KEY,
    -> customer_id INT,
    -> event_id INT,
    -> num_tickets INT,
    -> total_cost DECIMAL(10, 2),
    -> booking_date DATE,
    -> FOREIGN KEY (customer_id) REFERENCES Customer(customer_id),
    -> FOREIGN KEY (event_id) REFERENCES Event(event_id)
    -> );
Query OK, 0 rows affected (0.12 sec)
```

3. Create an ERD (Entity Relationship Diagram) for the database.



4. Create appropriate Primary Key and Foreign Key constraints for referential integrity

<u>**Venu Table:**</u>

Primary Key: venue_id :ensuring each venue has a unique identifier.

<u>**Event Table:**</u>

Primary Key: event_id uniquely identifies each event.

Foreign Key: venue_id references the primary key venue_id in the "Venu" table..

<u>**Customer Table:**</u>

Primary Key: customer_id uniquely identifies each customer.

Foreign Key: booking_id references the primary key booking_id in the "Booking" table, connecting customers to their bookings.

<u>**Booking Table:**</u>

Primary Key: booking_id uniquely identifies each booking.

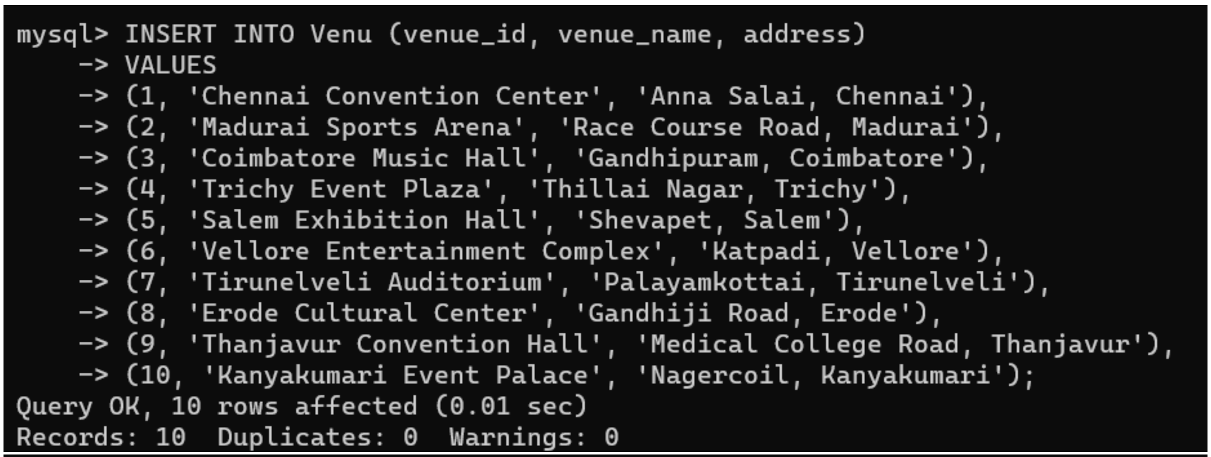
Foreign Keys:

customer_id references the primary key customer_id in the "Customer" table, establishing a link between bookings and customers.

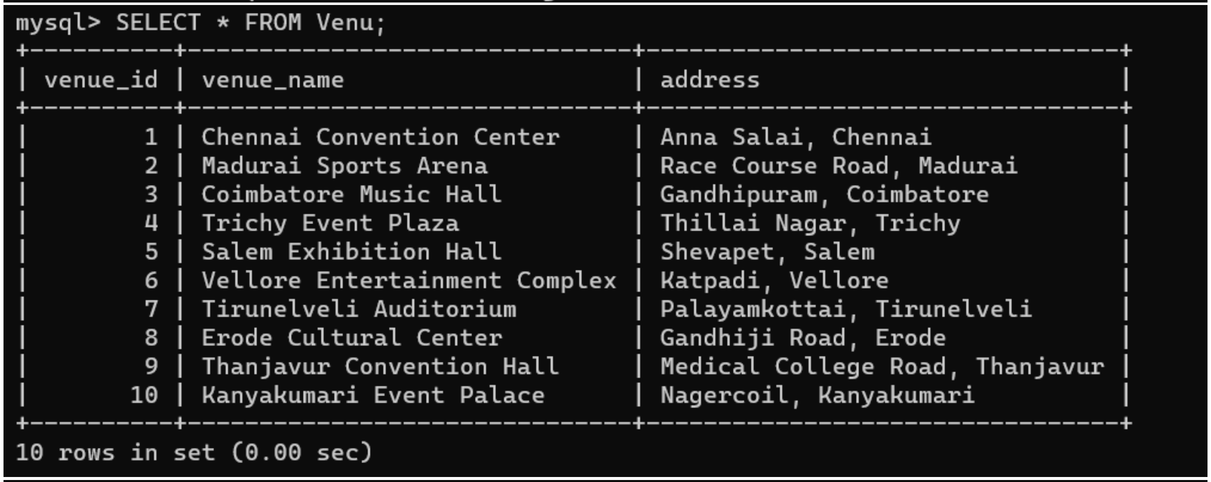
event_id references the primary key event_id in the "Event" table, linking bookings to specific events.

Tasks 2: Select, Where, Between, AND, LIKE:

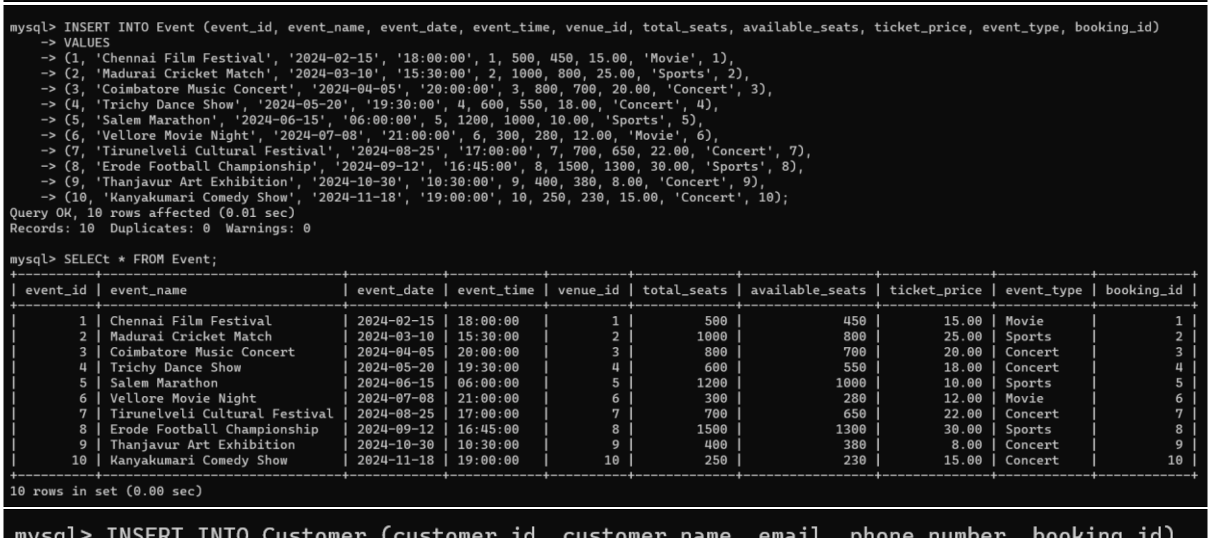
1. Write a SQL query to insert at least 10 sample records into each table.



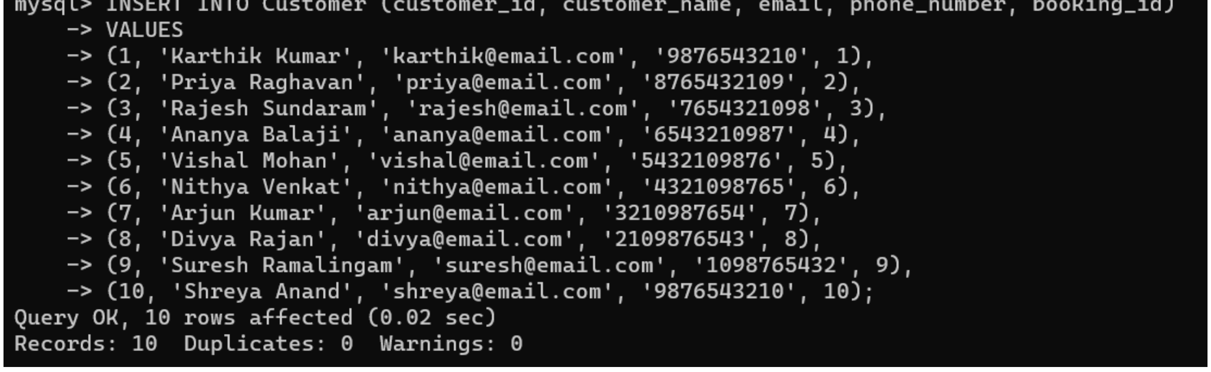
```
mysql> INSERT INTO Venu (venue_id, venue_name, address)
    -> VALUES
    -> (1, 'Chennai Convention Center', 'Anna Salai, Chennai'),
    -> (2, 'Madurai Sports Arena', 'Race Course Road, Madurai'),
    -> (3, 'Coimbatore Music Hall', 'Gandhipuram, Coimbatore'),
    -> (4, 'Trichy Event Plaza', 'Thillai Nagar, Trichy'),
    -> (5, 'Salem Exhibition Hall', 'Shevapet, Salem'),
    -> (6, 'Vellore Entertainment Complex', 'Katpadi, Vellore'),
    -> (7, 'Tirunelveli Auditorium', 'Palayamkottai, Tirunelveli'),
    -> (8, 'Erode Cultural Center', 'Gandhiji Road, Erode'),
    -> (9, 'Thanjavur Convention Hall', 'Medical College Road, Thanjavur'),
    -> (10, 'Kanyakumari Event Palace', 'Nagercoil, Kanyakumari');
Query OK, 10 rows affected (0.01 sec)
Records: 10  Duplicates: 0  Warnings: 0
```



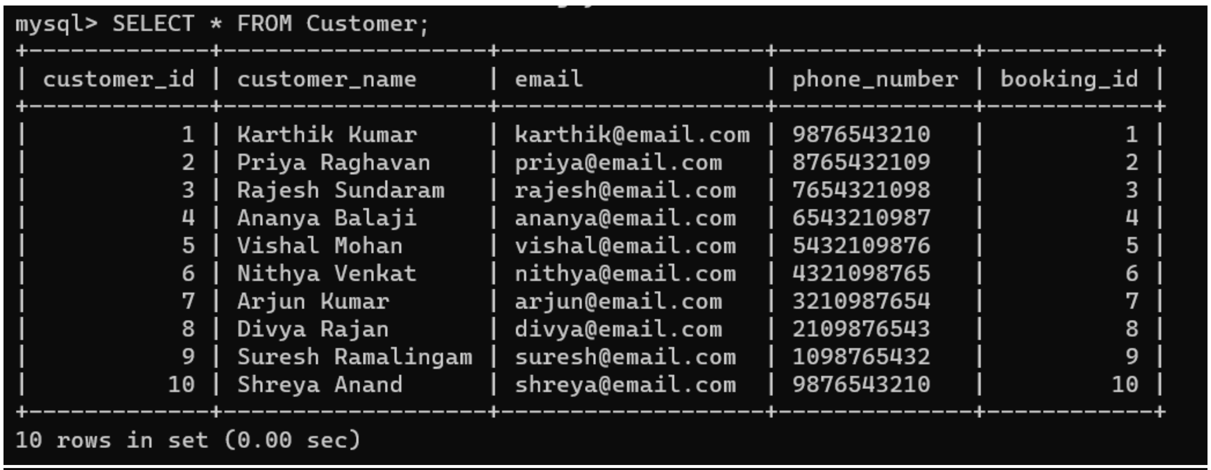
```
mysql> SELECT * FROM Venu;
+----------+-------------------------------+---------------------------------+
| venue_id | venue_name                    | address                         |
+----------+-------------------------------+---------------------------------+
|        1 | Chennai Convention Center     | Anna Salai, Chennai             |
|        2 | Madurai Sports Arena          | Race Course Road, Madurai       |
|        3 | Coimbatore Music Hall         | Gandhipuram, Coimbatore         |
|        4 | Trichy Event Plaza            | Thillai Nagar, Trichy           |
|        5 | Salem Exhibition Hall         | Shevapet, Salem                 |
|        6 | Vellore Entertainment Complex | Katpadi, Vellore                |
|        7 | Tirunelveli Auditorium        | Palayamkottai, Tirunelveli      |
|        8 | Erode Cultural Center         | Gandhiji Road, Erode            |
|        9 | Thanjavur Convention Hall     | Medical College Road, Thanjavur |
|       10 | Kanyakumari Event Palace      | Nagercoil, Kanyakumari          |
+----------+-------------------------------+---------------------------------+
10 rows in set (0.00 sec)
```



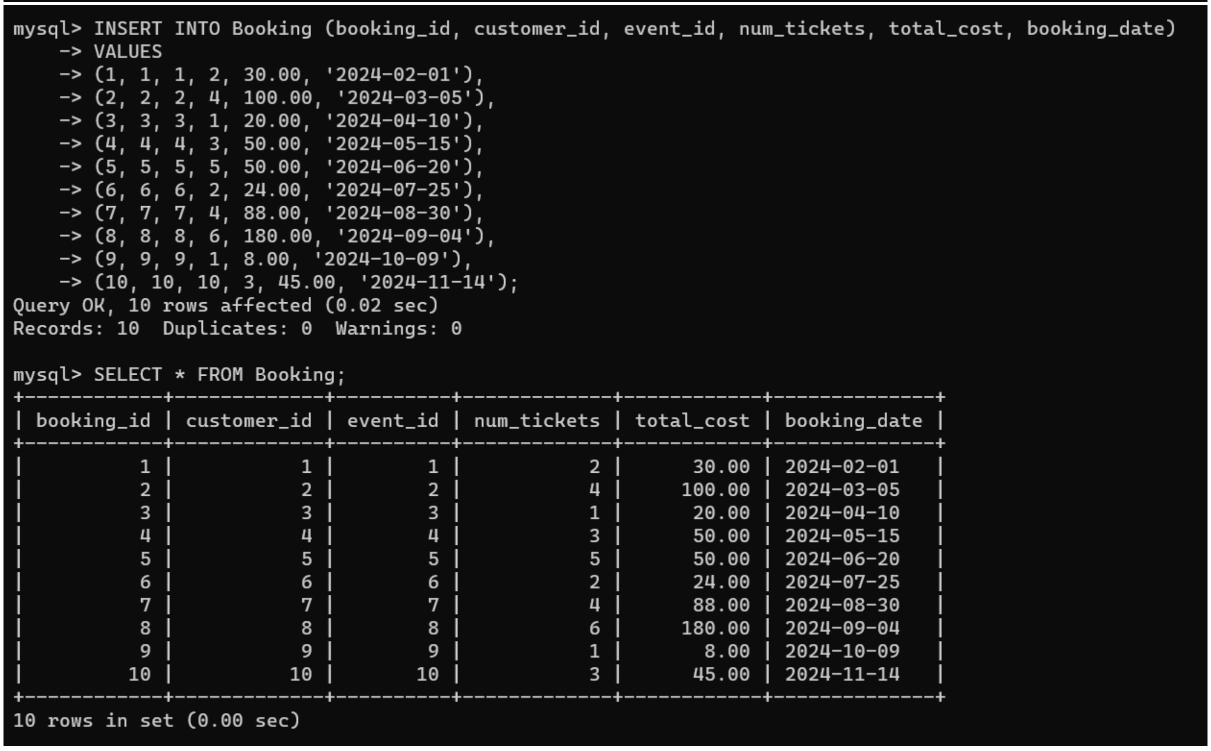
```
mysql> INSERT INTO Event (event_id, event_name, event_date, event_time, venue_id, total_seats, available_seats, ticket_price, event_type, booking_id)
    -> VALUES
    -> (1, 'Chennai Film Festival', '2024-02-15', '18:00:00', 1, 500, 450, 15.00, 'Movie', 1),
    -> (2, 'Madurai Cricket Match', '2024-03-10', '15:30:00', 2, 1000, 800, 25.00, 'Sports', 2),
    -> (3, 'Coimbatore Music Concert', '2024-04-05', '20:00:00', 3, 800, 700, 20.00, 'Concert', 3),
    -> (4, 'Trichy Dance Show', '2024-05-20', '19:30:00', 4, 600, 550, 18.00, 'Concert', 4),
    -> (5, 'Salem Marathon', '2024-06-15', '06:00:00', 5, 1200, 1000, 10.00, 'Sports', 5),
    -> (6, 'Vellore Movie Night', '2024-07-08', '21:00:00', 6, 300, 280, 12.00, 'Movie', 6),
    -> (7, 'Tirunelveli Cultural Festival', '2024-08-25', '17:00:00', 7, 700, 650, 22.00, 'Concert', 7),
    -> (8, 'Erode Football Championship', '2024-09-12', '16:45:00', 8, 1500, 1300, 30.00, 'Sports', 8),
    -> (9, 'Thanjavur Art Exhibition', '2024-10-30', '10:30:00', 9, 400, 380, 8.00, 'Concert', 9),
    -> (10, 'Kanyakumari Comedy Show', '2024-11-18', '19:00:00', 10, 250, 230, 15.00, 'Concert', 10);
Query OK, 10 rows affected (0.01 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> SELECt * FROM Event;
+----------+-------------------------------+------------+------------+----------+-------------+-----------------+--------------+------------+------------+
| event_id | event_name                    | event_date | event_time | venue_id | total_seats | available_seats | ticket_price | event_type | booking_id |
+----------+-------------------------------+------------+------------+----------+-------------+-----------------+--------------+------------+------------+
|        1 | Chennai Film Festival         | 2024-02-15 | 18:00:00   |        1 |         500 |             450 |        15.00 | Movie      |          1 |
|        2 | Madurai Cricket Match         | 2024-03-10 | 15:30:00   |        2 |        1000 |             800 |        25.00 | Sports     |          2 |
|        3 | Coimbatore Music Concert      | 2024-04-05 | 20:00:00   |        3 |         800 |             700 |        20.00 | Concert    |          3 |
|        4 | Trichy Dance Show             | 2024-05-20 | 19:30:00   |        4 |         600 |             550 |        18.00 | Concert    |          4 |
|        5 | Salem Marathon                | 2024-06-15 | 06:00:00   |        5 |        1200 |            1000 |        10.00 | Sports     |          5 |
|        6 | Vellore Movie Night           | 2024-07-08 | 21:00:00   |        6 |         300 |             280 |        12.00 | Movie      |          6 |
|        7 | Tirunelveli Cultural Festival | 2024-08-25 | 17:00:00   |        7 |         700 |             650 |        22.00 | Concert    |          7 |
|        8 | Erode Football Championship   | 2024-09-12 | 16:45:00   |        8 |        1500 |            1300 |        30.00 | Sports     |          8 |
|        9 | Thanjavur Art Exhibition      | 2024-10-30 | 10:30:00   |        9 |         400 |             380 |         8.00 | Concert    |          9 |
|       10 | Kanyakumari Comedy Show       | 2024-11-18 | 19:00:00   |       10 |         250 |             230 |        15.00 | Concert    |         10 |
+----------+-------------------------------+------------+------------+----------+-------------+-----------------+--------------+------------+------------+
10 rows in set (0.00 sec)
```



```
mysql> INSERT INTO Customer (customer_id, customer_name, email, phone_number, booking_id)
    -> VALUES
    -> (1, 'Karthik Kumar', 'karthik@email.com', '9876543210', 1),
    -> (2, 'Priya Raghavan', 'priya@email.com', '8765432109', 2),
    -> (3, 'Rajesh Sundaram', 'rajesh@email.com', '7654321098', 3),
    -> (4, 'Ananya Balaji', 'ananya@email.com', '6543210987', 4),
    -> (5, 'Vishal Mohan', 'vishal@email.com', '5432109876', 5),
    -> (6, 'Nithya Venkat', 'nithya@email.com', '4321098765', 6),
    -> (7, 'Arjun Kumar', 'arjun@email.com', '3210987654', 7),
    -> (8, 'Divya Rajan', 'divya@email.com', '2109876543', 8),
    -> (9, 'Suresh Ramalingam', 'suresh@email.com', '1098765432', 9),
    -> (10, 'Shreya Anand', 'shreya@email.com', '9876543210', 10);
Query OK, 10 rows affected (0.02 sec)
Records: 10  Duplicates: 0  Warnings: 0
```

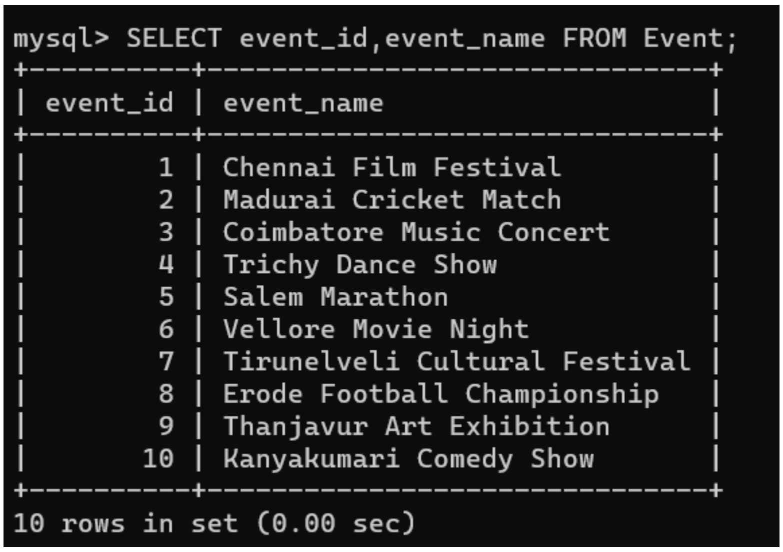


```
mysql> SELECT * FROM Customer;
+-------------+--------------------+-------------------+--------------+------------+
| customer_id | customer_name      | email             | phone_number | booking_id |
+-------------+--------------------+-------------------+--------------+------------+
|           1 | Karthik Kumar      | karthik@email.com | 9876543210   |          1 |
|           2 | Priya Raghavan     | priya@email.com   | 8765432109   |          2 |
|           3 | Rajesh Sundaram    | rajesh@email.com  | 7654321098   |          3 |
|           4 | Ananya Balaji      | ananya@email.com  | 6543210987   |          4 |
|           5 | Vishal Mohan       | vishal@email.com  | 5432109876   |          5 |
|           6 | Nithya Venkat      | nithya@email.com  | 4321098765   |          6 |
|           7 | Arjun Kumar        | arjun@email.com   | 3210987654   |          7 |
|           8 | Divya Rajan        | divya@email.com   | 2109876543   |          8 |
|           9 | Suresh Ramalingam  | suresh@email.com  | 1098765432   |          9 |
|          10 | Shreya Anand       | shreya@email.com  | 9876543210   |         10 |
+-------------+--------------------+-------------------+--------------+------------+
10 rows in set (0.00 sec)
```



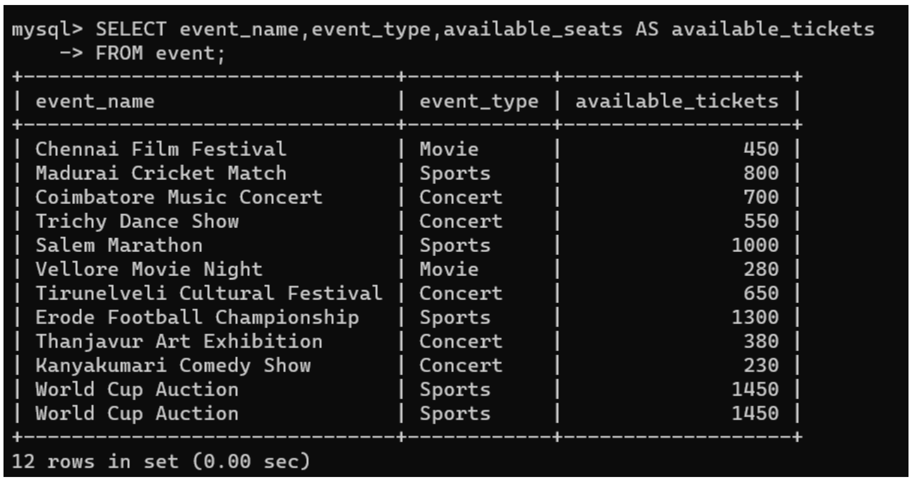
```
mysql> INSERT INTO Booking (booking_id, customer_id, event_id, num_tickets, total_cost, booking_date)
    -> VALUES
    -> (1, 1, 1, 2, 30.00, '2024-02-01'),
    -> (2, 2, 2, 4, 100.00, '2024-03-05'),
    -> (3, 3, 3, 1, 20.00, '2024-04-10'),
    -> (4, 4, 4, 3, 50.00, '2024-05-15'),
    -> (5, 5, 5, 5, 50.00, '2024-06-20'),
    -> (6, 6, 6, 2, 24.00, '2024-07-25'),
    -> (7, 7, 7, 4, 88.00, '2024-08-30'),
    -> (8, 8, 8, 6, 180.00, '2024-09-04'),
    -> (9, 9, 9, 1, 8.00, '2024-10-09'),
    -> (10, 10, 10, 3, 45.00, '2024-11-14');
Query OK, 10 rows affected (0.02 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM Booking;
+------------+-------------+----------+-------------+------------+--------------+
| booking_id | customer_id | event_id | num_tickets | total_cost | booking_date |
+------------+-------------+----------+-------------+------------+--------------+
|          1 |           1 |        1 |           2 |      30.00 | 2024-02-01   |
|          2 |           2 |        2 |           4 |     100.00 | 2024-03-05   |
|          3 |           3 |        3 |           1 |      20.00 | 2024-04-10   |
|          4 |           4 |        4 |           3 |      50.00 | 2024-05-15   |
|          5 |           5 |        5 |           5 |      50.00 | 2024-06-20   |
|          6 |           6 |        6 |           2 |      24.00 | 2024-07-25   |
|          7 |           7 |        7 |           4 |      88.00 | 2024-08-30   |
|          8 |           8 |        8 |           6 |     180.00 | 2024-09-04   |
|          9 |           9 |        9 |           1 |       8.00 | 2024-10-09   |
|         10 |          10 |       10 |           3 |      45.00 | 2024-11-14   |
+------------+-------------+----------+-------------+------------+--------------+
10 rows in set (0.00 sec)
```

2. Write a SQL query to list all Events.



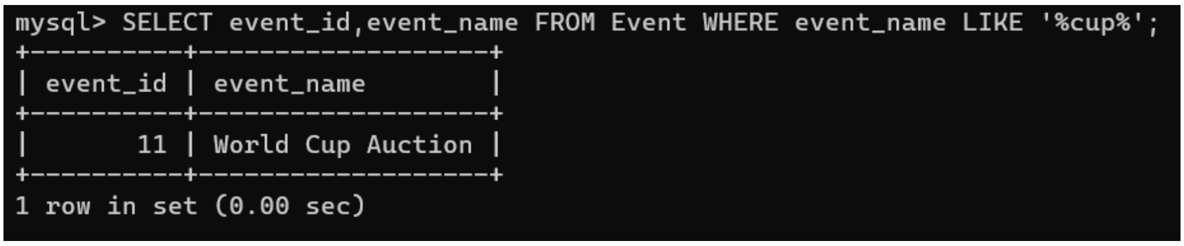
```
mysql> SELECT event_id,event_name FROM Event;
+----------+--------------------------------+
| event_id | event_name                     |
+----------+--------------------------------+
|        1 | Chennai Film Festival          |
|        2 | Madurai Cricket Match          |
|        3 | Coimbatore Music Concert       |
|        4 | Trichy Dance Show              |
|        5 | Salem Marathon                 |
|        6 | Vellore Movie Night            |
|        7 | Tirunelveli Cultural Festival  |
|        8 | Erode Football Championship    |
|        9 | Thanjavur Art Exhibition       |
|       10 | Kanyakumari Comedy Show        |
+----------+--------------------------------+
10 rows in set (0.00 sec)
```

3. Write a SQL query to select events with available tickets.



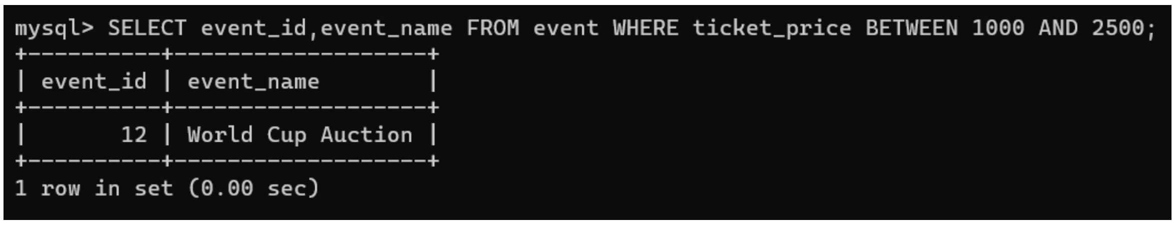
```
mysql> SELECT event_name,event_type,available_seats AS available_tickets
    -> FROM event;
+-------------------------------+------------+-------------------+
| event_name                    | event_type | available_tickets |
+-------------------------------+------------+-------------------+
| Chennai Film Festival         | Movie      |               450 |
| Madurai Cricket Match         | Sports     |               800 |
| Coimbatore Music Concert      | Concert    |               700 |
| Trichy Dance Show             | Concert    |               550 |
| Salem Marathon                | Sports     |              1000 |
| Vellore Movie Night           | Movie      |               280 |
| Tirunelveli Cultural Festival | Concert    |               650 |
| Erode Football Championship   | Sports     |              1300 |
| Thanjavur Art Exhibition      | Concert    |               380 |
| Kanyakumari Comedy Show       | Concert    |               230 |
| World Cup Auction             | Sports     |              1450 |
| World Cup Auction             | Sports     |              1450 |
+-------------------------------+------------+-------------------+
12 rows in set (0.00 sec)
```

4. Write a SQL query to select events name partial match with 'cup'.



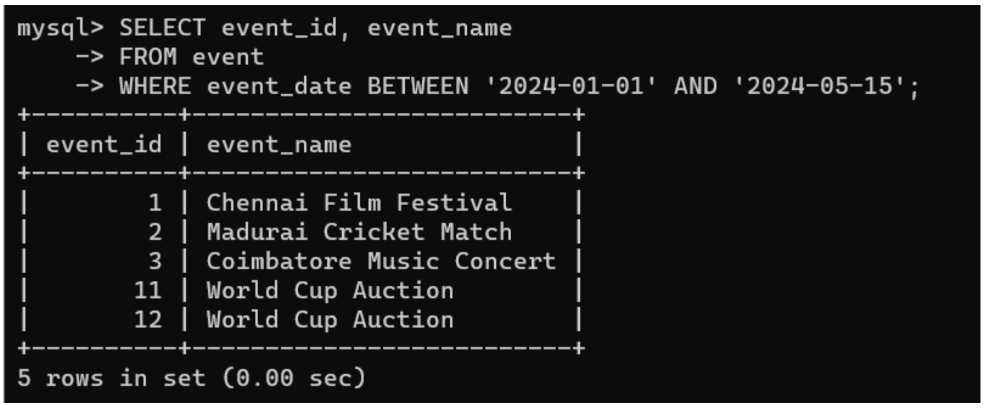
```
mysql> SELECT event_id,event_name FROM Event WHERE event_name LIKE '%cup%';
+----------+-------------------+
| event_id | event_name        |
+----------+-------------------+
|       11 | World Cup Auction |
+----------+-------------------+
1 row in set (0.00 sec)
```

5. Write a SQL query to select events with ticket price range is between 1000 to 2500.



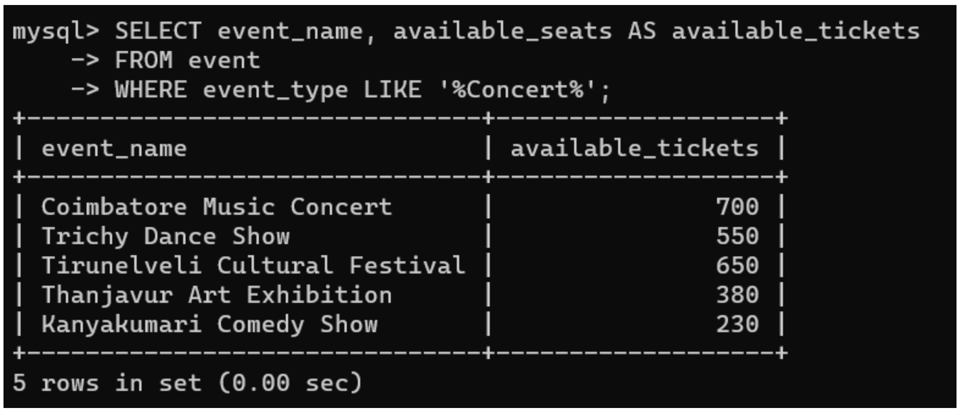
```
mysql> SELECT event_id,event_name FROM event WHERE ticket_price BETWEEN 1000 AND 2500;
+----------+-------------------+
| event_id | event_name        |
+----------+-------------------+
|       12 | World Cup Auction |
+----------+-------------------+
1 row in set (0.00 sec)
```

6. Write a SQL query to retrieve events with dates falling within a specific range.



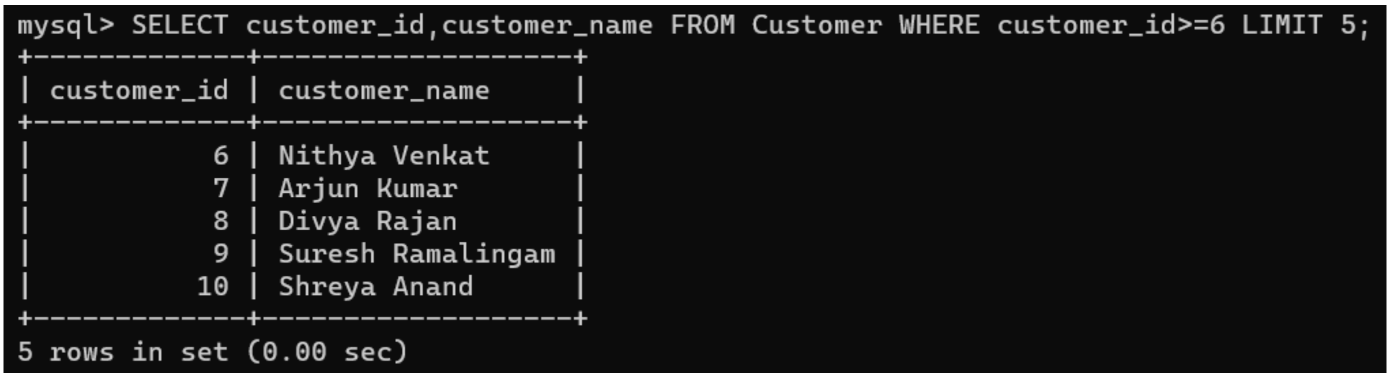
```
mysql> SELECT event_id, event_name
    -> FROM event
    -> WHERE event_date BETWEEN '2024-01-01' AND '2024-05-15';
+----------+--------------------------+
| event_id | event_name               |
+----------+--------------------------+
|        1 | Chennai Film Festival    |
|        2 | Madurai Cricket Match    |
|        3 | Coimbatore Music Concert |
|       11 | World Cup Auction        |
|       12 | World Cup Auction        |
+----------+--------------------------+
5 rows in set (0.00 sec)
```

7. Write a SQL query to retrieve events with available tickets that also have "Concert" in their name.



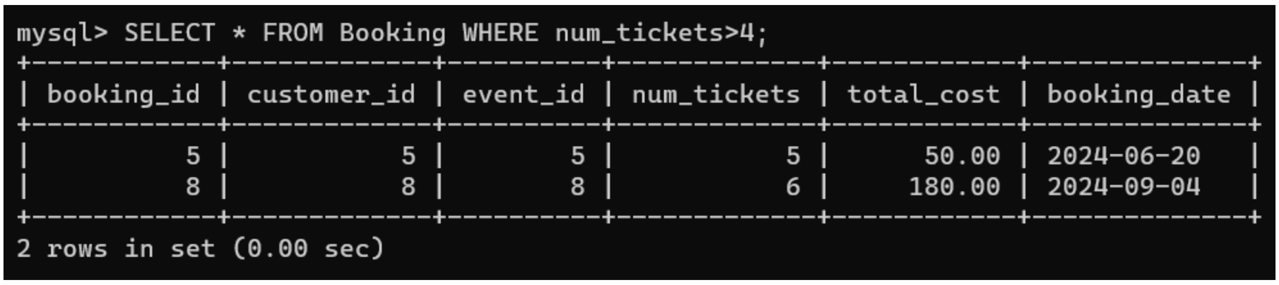
```
mysql> SELECT event_name, available_seats AS available_tickets
    -> FROM event
    -> WHERE event_type LIKE '%Concert%';
+-------------------------------+-------------------+
| event_name                    | available_tickets |
+-------------------------------+-------------------+
| Coimbatore Music Concert      |               700 |
| Trichy Dance Show             |               550 |
| Tirunelveli Cultural Festival |               650 |
| Thanjavur Art Exhibition      |               380 |
| Kanyakumari Comedy Show       |               230 |
+-------------------------------+-------------------+
5 rows in set (0.00 sec)
```

8. Write a SQL query to retrieve users in batches of 5, starting from the 6th user.



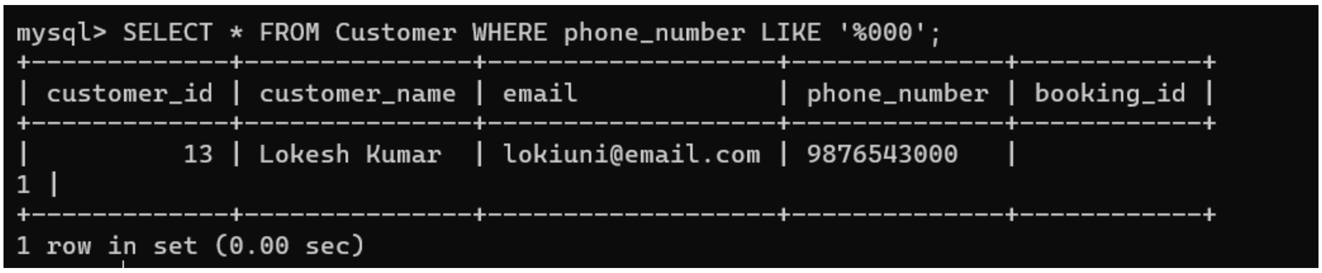
```
mysql> SELECT customer_id,customer_name FROM Customer WHERE customer_id>=6 LIMIT 5;
+-------------+-------------------+
| customer_id | customer_name     |
+-------------+-------------------+
|           6 | Nithya Venkat     |
|           7 | Arjun Kumar       |
|           8 | Divya Rajan       |
|           9 | Suresh Ramalingam |
|          10 | Shreya Anand      |
+-------------+-------------------+
5 rows in set (0.00 sec)
```

9. Write a SQL query to retrieve bookings details contains booked no of ticket more than 4.



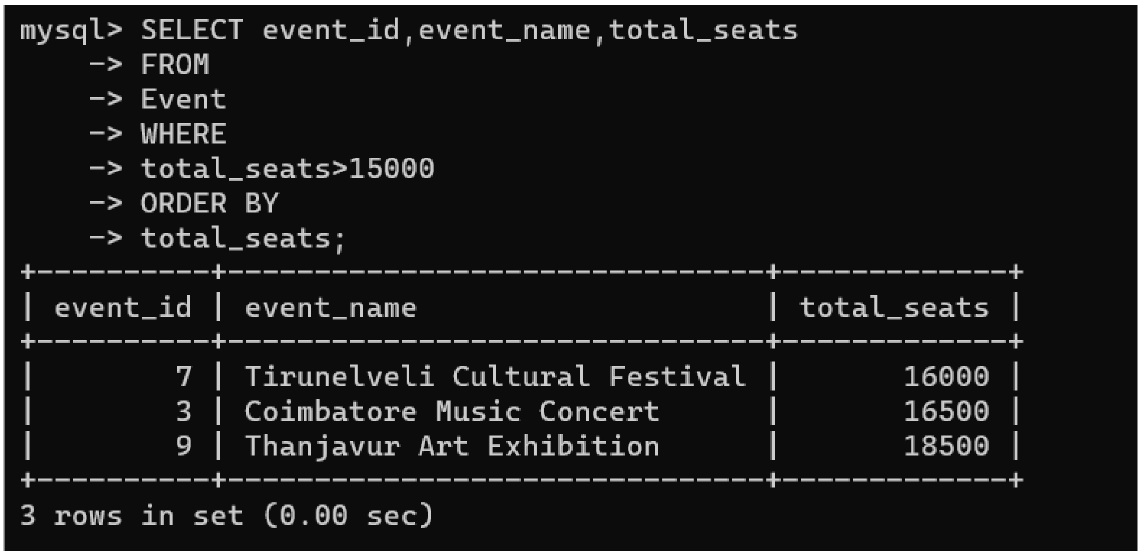
```
mysql> SELECT * FROM Booking WHERE num_tickets>4;
+------------+-------------+----------+-------------+------------+--------------+
| booking_id | customer_id | event_id | num_tickets | total_cost | booking_date |
+------------+-------------+----------+-------------+------------+--------------+
|          5 |           5 |        5 |           5 |      50.00 | 2024-06-20   |
|          8 |           8 |        8 |           6 |     180.00 | 2024-09-04   |
+------------+-------------+----------+-------------+------------+--------------+
2 rows in set (0.00 sec)
```

10. Write a SQL query to retrieve customer information whose phone number end with '000'



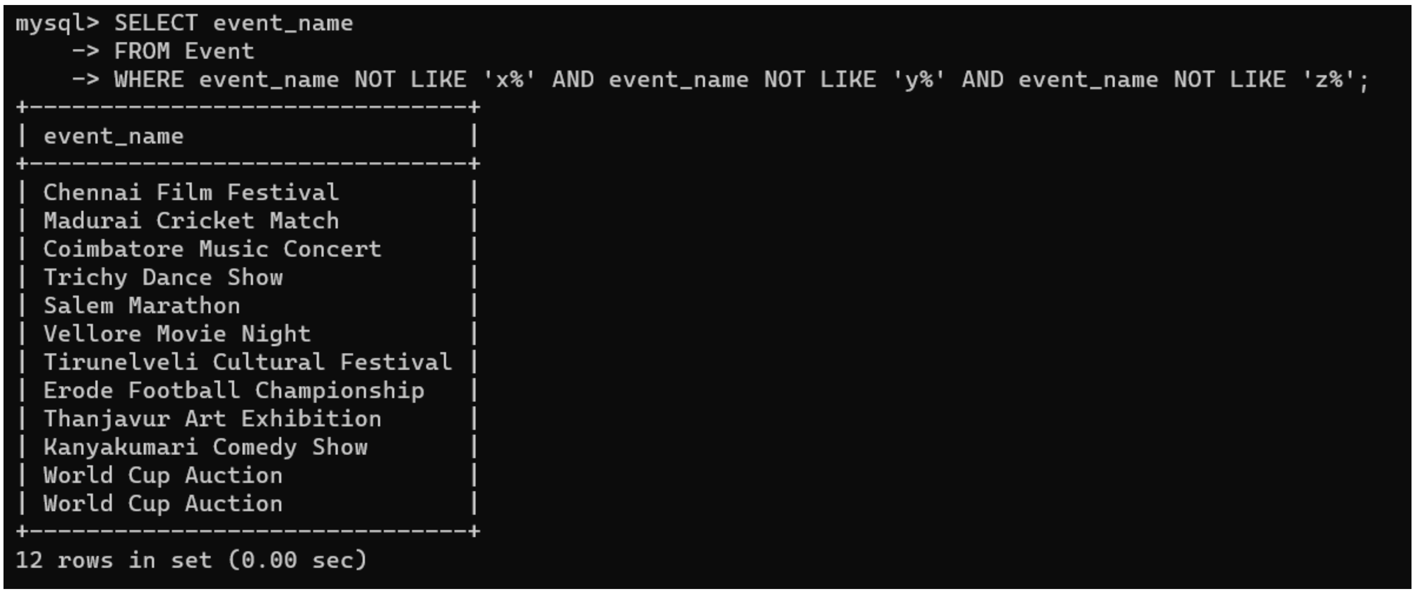
```
mysql> SELECT * FROM Customer WHERE phone_number LIKE '%000';
+-------------+---------------+-------------------+--------------+------------+
| customer_id | customer_name | email             | phone_number | booking_id |
+-------------+---------------+-------------------+--------------+------------+
|          13 | Lokesh Kumar  | lokiuni@email.com | 9876543000   |
1 |
+-------------+---------------+-------------------+--------------+------------+
1 row in set (0.00 sec)
```

11. Write a SQL query to retrieve the events in order whose seat capacity more than 15000.



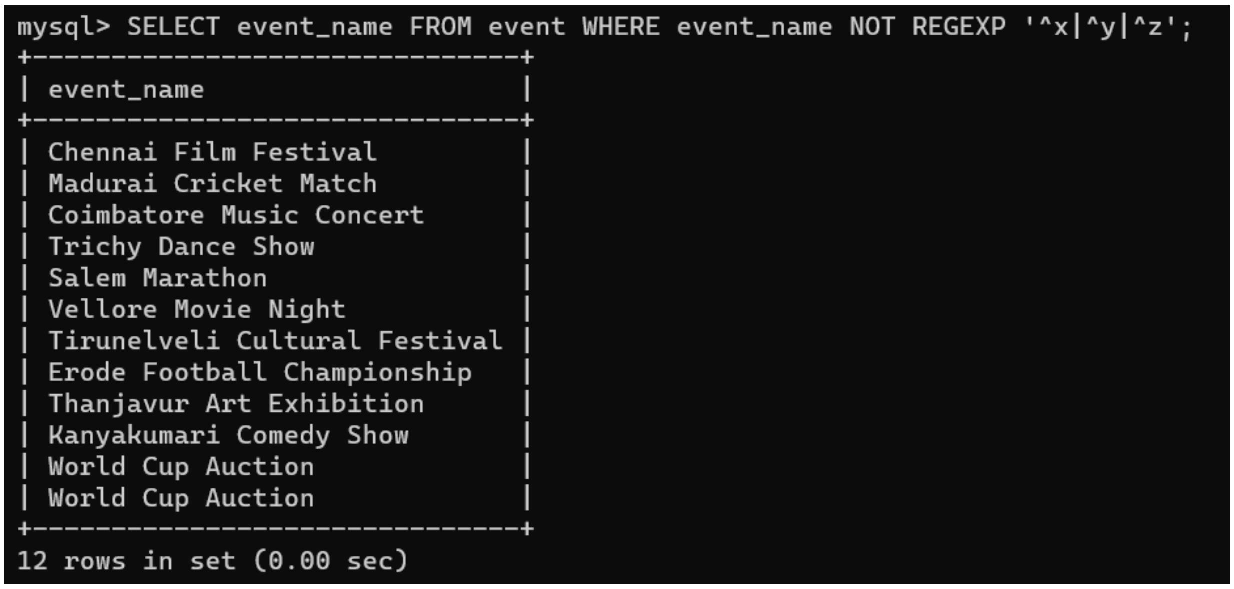
```
mysql> SELECT event_id,event_name,total_seats
    -> FROM
    -> Event
    -> WHERE
    -> total_seats>15000
    -> ORDER BY
    -> total_seats;
+----------+--------------------------------+-------------+
| event_id | event_name                     | total_seats |
+----------+--------------------------------+-------------+
|        7 | Tirunelveli Cultural Festival  |       16000 |
|        3 | Coimbatore Music Concert       |       16500 |
|        9 | Thanjavur Art Exhibition       |       18500 |
+----------+--------------------------------+-------------+
3 rows in set (0.00 sec)
```

12. Write a SQL query to select events name not start with ‘x’, ‘y’, ‘z’



```
mysql> SELECT event_name
    -> FROM Event
    -> WHERE event_name NOT LIKE 'x%' AND event_name NOT LIKE 'y%' AND event_name NOT LIKE 'z%';
+--------------------------------+
| event_name                     |
+--------------------------------+
| Chennai Film Festival          |
| Madurai Cricket Match          |
| Coimbatore Music Concert       |
| Trichy Dance Show              |
| Salem Marathon                 |
| Vellore Movie Night            |
| Tirunelveli Cultural Festival  |
| Erode Football Championship    |
| Thanjavur Art Exhibition       |
| Kanyakumari Comedy Show        |
| World Cup Auction              |
| World Cup Auction              |
+--------------------------------+
12 rows in set (0.00 sec)
```

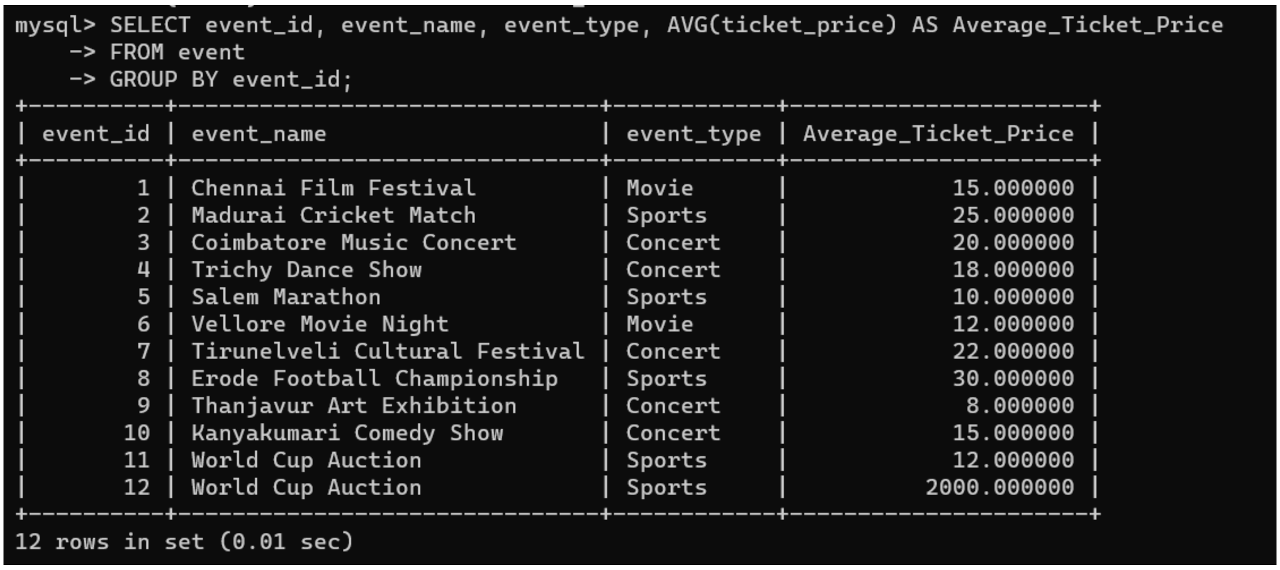
(or)



```
mysql> SELECT event_name FROM event WHERE event_name NOT REGEXP '^x|^y|^z';
+--------------------------------+
| event_name                     |
+--------------------------------+
| Chennai Film Festival          |
| Madurai Cricket Match          |
| Coimbatore Music Concert       |
| Trichy Dance Show              |
| Salem Marathon                 |
| Vellore Movie Night            |
| Tirunelveli Cultural Festival  |
| Erode Football Championship    |
| Thanjavur Art Exhibition       |
| Kanyakumari Comedy Show        |
| World Cup Auction              |
| World Cup Auction              |
+--------------------------------+
12 rows in set (0.00 sec)
```

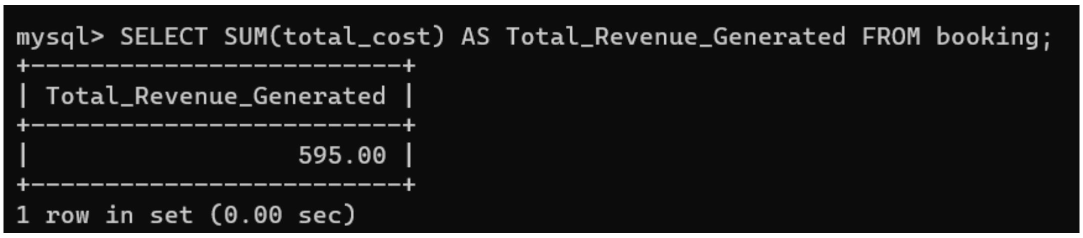
Tasks 3: Aggregate functions, Having, Order By, GroupBy and Joins:

1. Write a SQL query to List Events and Their Average Ticket Prices.



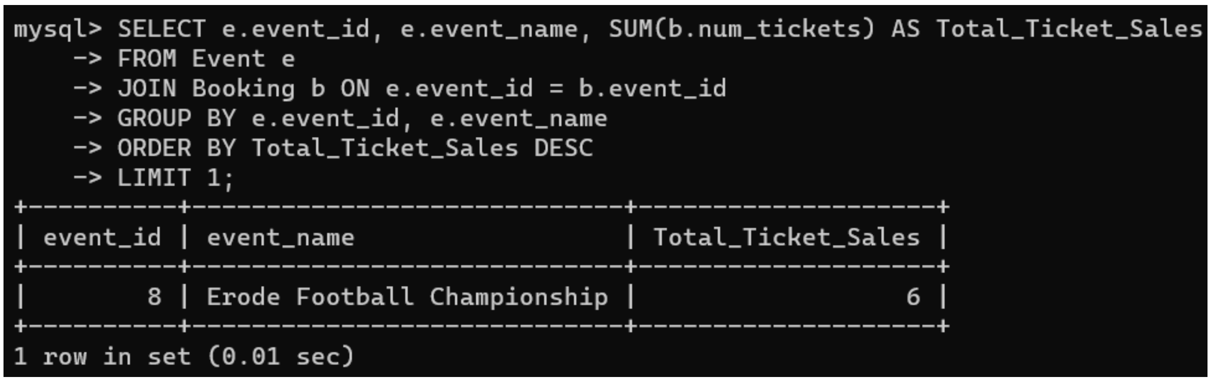
```
mysql> SELECT event_id, event_name, event_type, AVG(ticket_price) AS Average_Ticket_Price
    -> FROM event
    -> GROUP BY event_id;
+----------+-------------------------------+------------+----------------------+
| event_id | event_name                    | event_type | Average_Ticket_Price |
+----------+-------------------------------+------------+----------------------+
|        1 | Chennai Film Festival         | Movie      |            15.000000 |
|        2 | Madurai Cricket Match         | Sports     |            25.000000 |
|        3 | Coimbatore Music Concert      | Concert    |            20.000000 |
|        4 | Trichy Dance Show             | Concert    |            18.000000 |
|        5 | Salem Marathon                | Sports     |            10.000000 |
|        6 | Vellore Movie Night           | Movie      |            12.000000 |
|        7 | Tirunelveli Cultural Festival | Concert    |            22.000000 |
|        8 | Erode Football Championship   | Sports     |            30.000000 |
|        9 | Thanjavur Art Exhibition      | Concert    |             8.000000 |
|       10 | Kanyakumari Comedy Show       | Concert    |            15.000000 |
|       11 | World Cup Auction             | Sports     |            12.000000 |
|       12 | World Cup Auction             | Sports     |          2000.000000 |
+----------+-------------------------------+------------+----------------------+
12 rows in set (0.01 sec)
```

2. Write a SQL query to Calculate the Total Revenue Generated by Events.



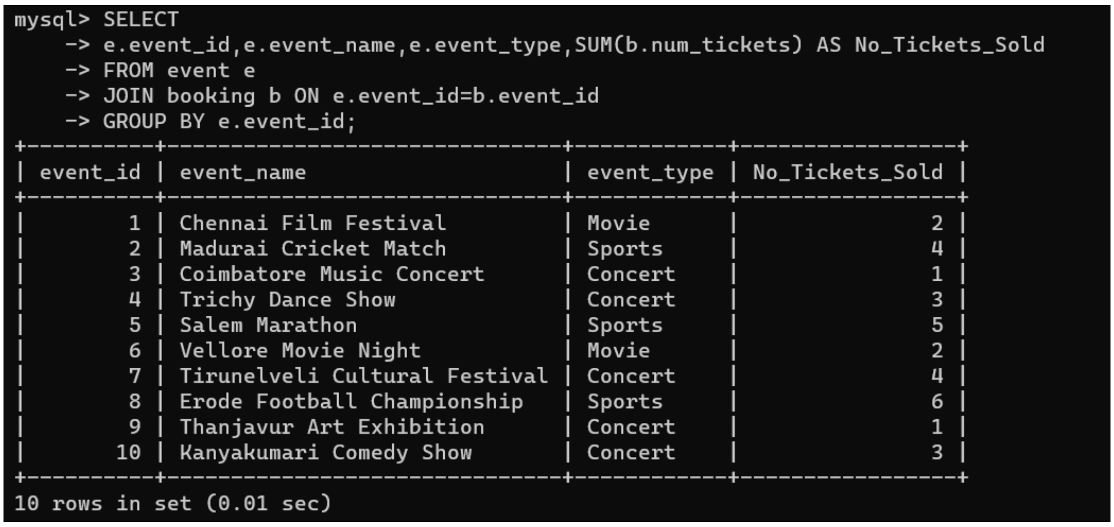
```
mysql> SELECT SUM(total_cost) AS Total_Revenue_Generated FROM booking;
+-------------------------+
| Total_Revenue_Generated |
+-------------------------+
|                  595.00 |
+-------------------------+
1 row in set (0.00 sec)
```

3. Write a SQL query to find the event with the highest ticket sales.



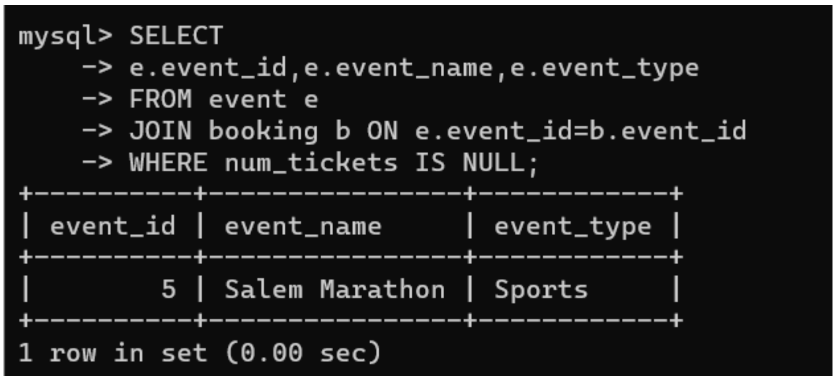
```
mysql> SELECT e.event_id, e.event_name, SUM(b.num_tickets) AS Total_Ticket_Sales
    -> FROM Event e
    -> JOIN Booking b ON e.event_id = b.event_id
    -> GROUP BY e.event_id, e.event_name
    -> ORDER BY Total_Ticket_Sales DESC
    -> LIMIT 1;
+----------+-----------------------------+--------------------+
| event_id | event_name                  | Total_Ticket_Sales |
+----------+-----------------------------+--------------------+
|        8 | Erode Football Championship |                  6 |
+----------+-----------------------------+--------------------+
1 row in set (0.01 sec)
```

4. Write a SQL query to Calculate the Total Number of Tickets Sold for Each Event.



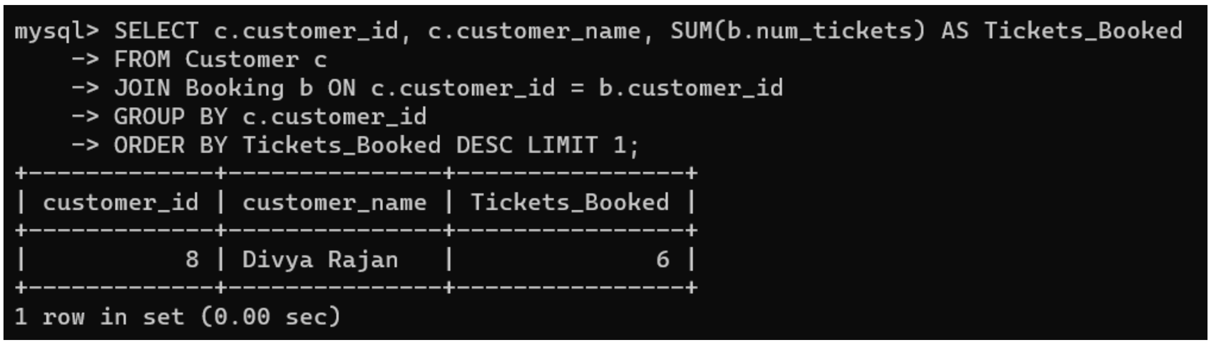
```
mysql> SELECT
    -> e.event_id,e.event_name,e.event_type,SUM(b.num_tickets) AS No_Tickets_Sold
    -> FROM event e
    -> JOIN booking b ON e.event_id=b.event_id
    -> GROUP BY e.event_id;
+----------+-------------------------------+------------+-----------------+
| event_id | event_name                    | event_type | No_Tickets_Sold |
+----------+-------------------------------+------------+-----------------+
|        1 | Chennai Film Festival         | Movie      |               2 |
|        2 | Madurai Cricket Match         | Sports     |               4 |
|        3 | Coimbatore Music Concert      | Concert    |               1 |
|        4 | Trichy Dance Show             | Concert    |               3 |
|        5 | Salem Marathon                | Sports     |               5 |
|        6 | Vellore Movie Night           | Movie      |               2 |
|        7 | Tirunelveli Cultural Festival | Concert    |               4 |
|        8 | Erode Football Championship   | Sports     |               6 |
|        9 | Thanjavur Art Exhibition      | Concert    |               1 |
|       10 | Kanyakumari Comedy Show       | Concert    |               3 |
+----------+-------------------------------+------------+-----------------+
10 rows in set (0.01 sec)
```

5. Write a SQL query to Find Events with No Ticket Sales.



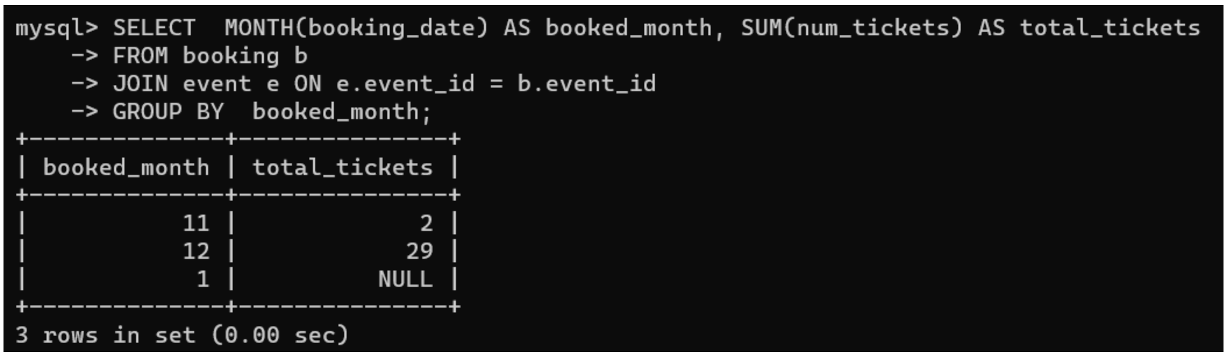
```
mysql> SELECT
    -> e.event_id,e.event_name,e.event_type
    -> FROM event e
    -> JOIN booking b ON e.event_id=b.event_id
    -> WHERE num_tickets IS NULL;
+----------+----------------+------------+
| event_id | event_name     | event_type |
+----------+----------------+------------+
|        5 | Salem Marathon | Sports     |
+----------+----------------+------------+
1 row in set (0.00 sec)
```

6. Write a SQL query to Find the User Who Has Booked the Most Tickets.



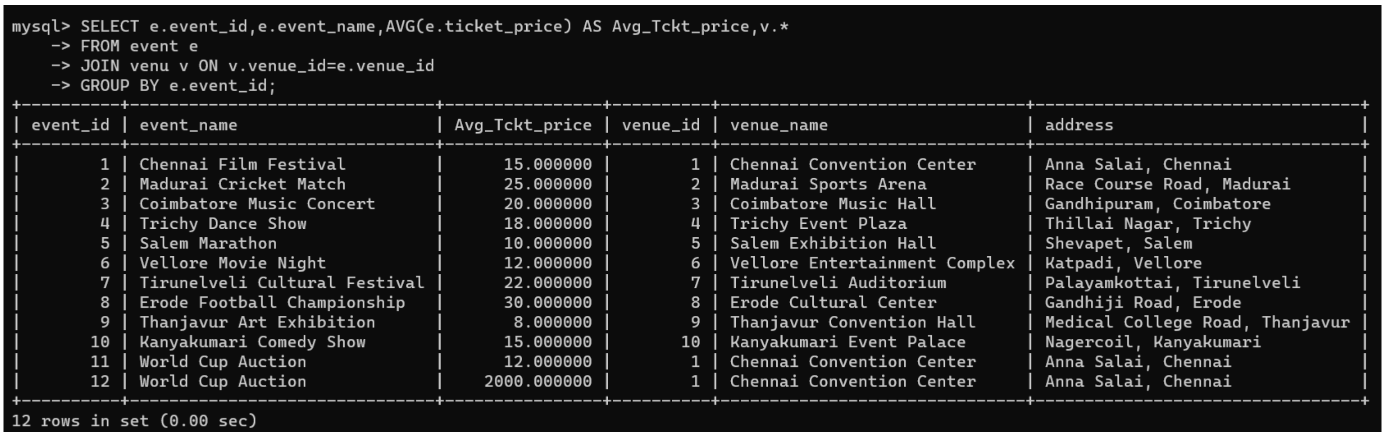
```
mysql> SELECT c.customer_id, c.customer_name, SUM(b.num_tickets) AS Tickets_Booked
    -> FROM Customer c
    -> JOIN Booking b ON c.customer_id = b.customer_id
    -> GROUP BY c.customer_id
    -> ORDER BY Tickets_Booked DESC LIMIT 1;
+-------------+---------------+----------------+
| customer_id | customer_name | Tickets_Booked |
+-------------+---------------+----------------+
|           8 | Divya Rajan   |              6 |
+-------------+---------------+----------------+
1 row in set (0.00 sec)
```

7. Write a SQL query to List Events and the total number of tickets sold for each month.



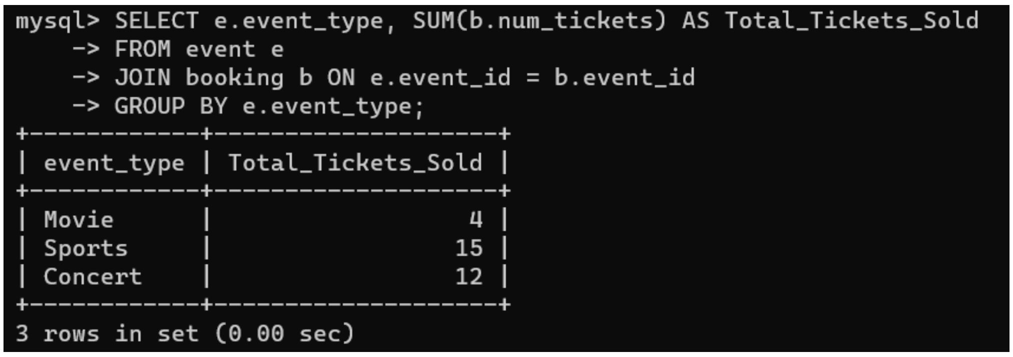
```
mysql> SELECT  MONTH(booking_date) AS booked_month, SUM(num_tickets) AS total_tickets
    -> FROM booking b
    -> JOIN event e ON e.event_id = b.event_id
    -> GROUP BY  booked_month;
+--------------+---------------+
| booked_month | total_tickets |
+--------------+---------------+
|           11 |             2 |
|           12 |            29 |
|            1 |          NULL |
+--------------+---------------+
3 rows in set (0.00 sec)
```

8. Write a SQL query to calculate the average Ticket Price for Events in Each Venue.



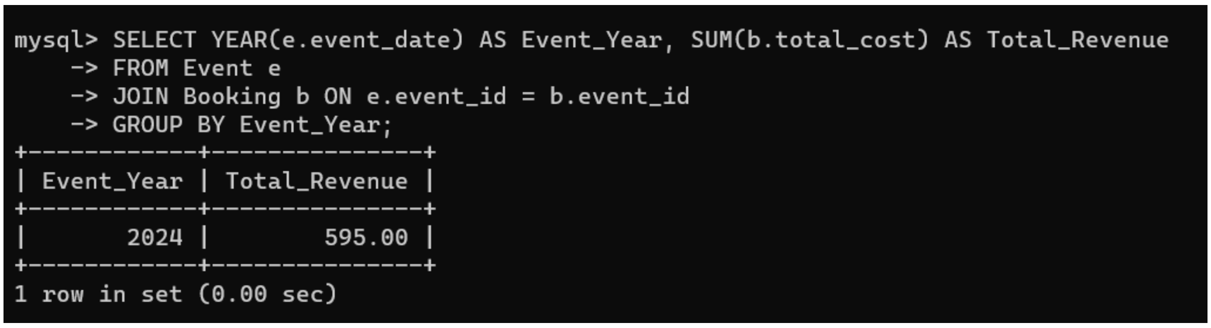
```
mysql> SELECT e.event_id,e.event_name,AVG(e.ticket_price) AS Avg_Tckt_price,v.*
    -> FROM event e
    -> JOIN venu v ON v.venue_id=e.venue_id
    -> GROUP BY e.event_id;
+----------+-------------------------------+----------------+----------+-------------------------------+---------------------------------+
| event_id | event_name                    | Avg_Tckt_price | venue_id | venue_name                    | address                         |
+----------+-------------------------------+----------------+----------+-------------------------------+---------------------------------+
|        1 | Chennai Film Festival         |      15.000000 |        1 | Chennai Convention Center     | Anna Salai, Chennai             |
|        2 | Madurai Cricket Match         |      25.000000 |        2 | Madurai Sports Arena          | Race Course Road, Madurai       |
|        3 | Coimbatore Music Concert      |      20.000000 |        3 | Coimbatore Music Hall         | Gandhipuram, Coimbatore         |
|        4 | Trichy Dance Show             |      18.000000 |        4 | Trichy Event Plaza            | Thillai Nagar, Trichy           |
|        5 | Salem Marathon                |      10.000000 |        5 | Salem Exhibition Hall         | Shevapet, Salem                 |
|        6 | Vellore Movie Night           |      12.000000 |        6 | Vellore Entertainment Complex | Katpadi, Vellore                |
|        7 | Tirunelveli Cultural Festival |      22.000000 |        7 | Tirunelveli Auditorium        | Palayamkottai, Tirunelveli      |
|        8 | Erode Football Championship   |      30.000000 |        8 | Erode Cultural Center         | Gandhiji Road, Erode            |
|        9 | Thanjavur Art Exhibition      |       8.000000 |        9 | Thanjavur Convention Hall     | Medical College Road, Thanjavur |
|       10 | Kanyakumari Comedy Show       |      15.000000 |       10 | Kanyakumari Event Palace      | Nagercoil, Kanyakumari          |
|       11 | World Cup Auction             |      12.000000 |        1 | Chennai Convention Center     | Anna Salai, Chennai             |
|       12 | World Cup Auction             |    2000.000000 |        1 | Chennai Convention Center     | Anna Salai, Chennai             |
+----------+-------------------------------+----------------+----------+-------------------------------+---------------------------------+
12 rows in set (0.00 sec)
```

9. Write a SQL query to calculate the total Number of Tickets Sold for Each Event Type.



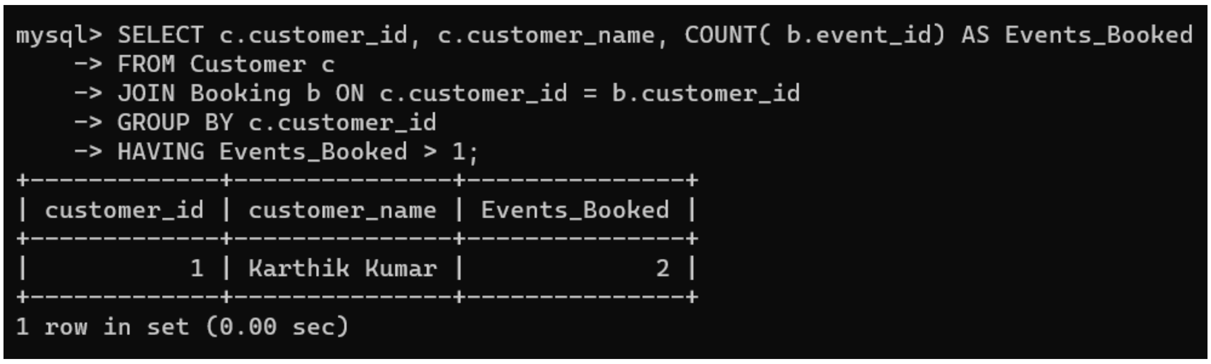
```
mysql> SELECT e.event_type, SUM(b.num_tickets) AS Total_Tickets_Sold
    -> FROM event e
    -> JOIN booking b ON e.event_id = b.event_id
    -> GROUP BY e.event_type;
+------------+--------------------+
| event_type | Total_Tickets_Sold |
+------------+--------------------+
| Movie      |                  4 |
| Sports     |                 15 |
| Concert    |                 12 |
+------------+--------------------+
3 rows in set (0.00 sec)
```

10. Write a SQL query to calculate the total Revenue Generated by Events in Each Year.



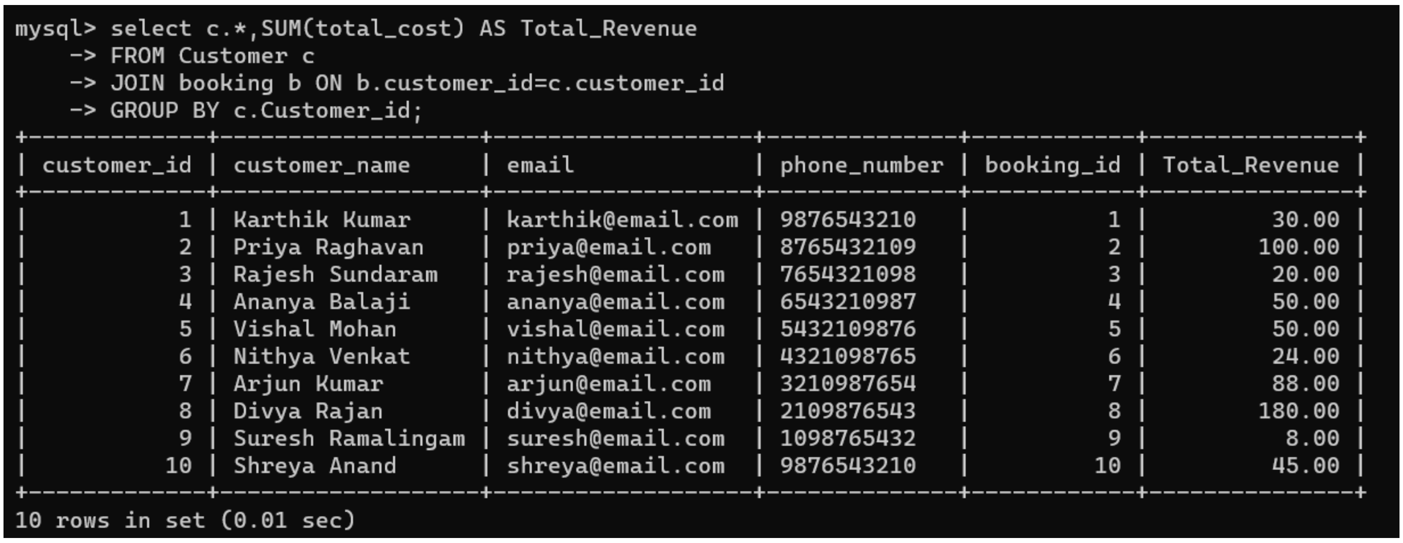
```
mysql> SELECT YEAR(e.event_date) AS Event_Year, SUM(b.total_cost) AS Total_Revenue
    -> FROM Event e
    -> JOIN Booking b ON e.event_id = b.event_id
    -> GROUP BY Event_Year;
+------------+---------------+
| Event_Year | Total_Revenue |
+------------+---------------+
|       2024 |        595.00 |
+------------+---------------+
1 row in set (0.00 sec)
```

11. Write a SQL query to list users who have booked tickets for multiple events.



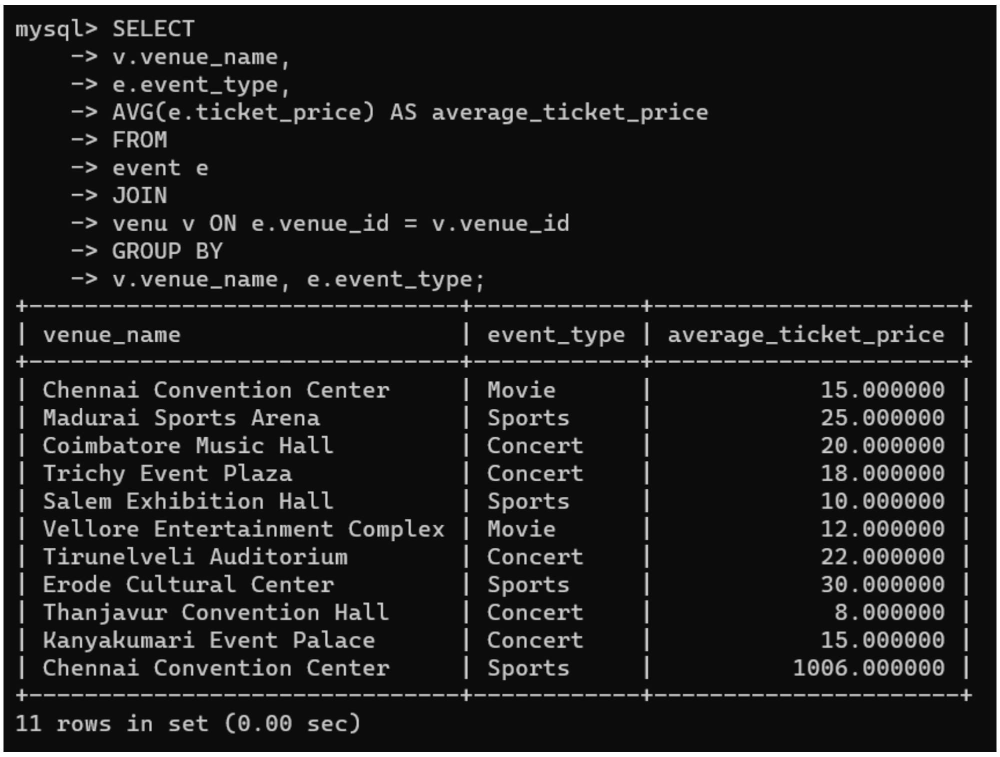
```
mysql> SELECT c.customer_id, c.customer_name, COUNT( b.event_id) AS Events_Booked
    -> FROM Customer c
    -> JOIN Booking b ON c.customer_id = b.customer_id
    -> GROUP BY c.customer_id
    -> HAVING Events_Booked > 1;
+-------------+---------------+---------------+
| customer_id | customer_name | Events_Booked |
+-------------+---------------+---------------+
|           1 | Karthik Kumar |             2 |
+-------------+---------------+---------------+
1 row in set (0.00 sec)
```

12. Write a SQL query to calculate the Total Revenue Generated by Events for Each User.



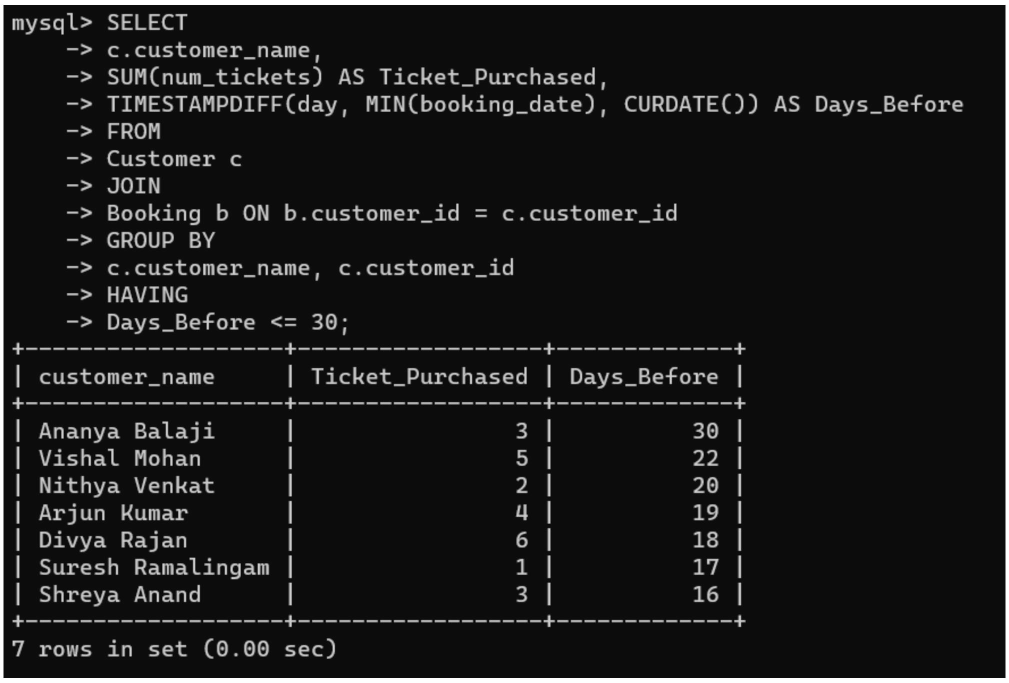
```
mysql> select c.*,SUM(total_cost) AS Total_Revenue
    -> FROM Customer c
    -> JOIN booking b ON b.customer_id=c.customer_id
    -> GROUP BY c.Customer_id;
+-------------+-------------------+-------------------+--------------+------------+---------------+
| customer_id | customer_name     | email             | phone_number | booking_id | Total_Revenue |
+-------------+-------------------+-------------------+--------------+------------+---------------+
|           1 | Karthik Kumar     | karthik@email.com | 9876543210   |          1 |         30.00 |
|           2 | Priya Raghavan    | priya@email.com   | 8765432109   |          2 |        100.00 |
|           3 | Rajesh Sundaram   | rajesh@email.com  | 7654321098   |          3 |         20.00 |
|           4 | Ananya Balaji     | ananya@email.com  | 6543210987   |          4 |         50.00 |
|           5 | Vishal Mohan      | vishal@email.com  | 5432109876   |          5 |         50.00 |
|           6 | Nithya Venkat     | nithya@email.com  | 4321098765   |          6 |         24.00 |
|           7 | Arjun Kumar       | arjun@email.com   | 3210987654   |          7 |         88.00 |
|           8 | Divya Rajan       | divya@email.com   | 2109876543   |          8 |        180.00 |
|           9 | Suresh Ramalingam | suresh@email.com  | 1098765432   |          9 |          8.00 |
|          10 | Shreya Anand      | shreya@email.com  | 9876543210   |         10 |         45.00 |
+-------------+-------------------+-------------------+--------------+------------+---------------+
10 rows in set (0.01 sec)
```

13. Write a SQL query to calculate the Average Ticket Price for Events in Each Category and Venue.



```
mysql> SELECT
    -> v.venue_name,
    -> e.event_type,
    -> AVG(e.ticket_price) AS average_ticket_price
    -> FROM
    -> event e
    -> JOIN
    -> venu v ON e.venue_id = v.venue_id
    -> GROUP BY
    -> v.venue_name, e.event_type;
+-------------------------------+------------+----------------------+
| venue_name                    | event_type | average_ticket_price |
+-------------------------------+------------+----------------------+
| Chennai Convention Center     | Movie      |            15.000000 |
| Madurai Sports Arena          | Sports     |            25.000000 |
| Coimbatore Music Hall         | Concert    |            20.000000 |
| Trichy Event Plaza            | Concert    |            18.000000 |
| Salem Exhibition Hall         | Sports     |            10.000000 |
| Vellore Entertainment Complex | Movie      |            12.000000 |
| Tirunelveli Auditorium        | Concert    |            22.000000 |
| Erode Cultural Center         | Sports     |            30.000000 |
| Thanjavur Convention Hall     | Concert    |             8.000000 |
| Kanyakumari Event Palace      | Concert    |            15.000000 |
| Chennai Convention Center     | Sports     |          1006.000000 |
+-------------------------------+------------+----------------------+
11 rows in set (0.00 sec)
```

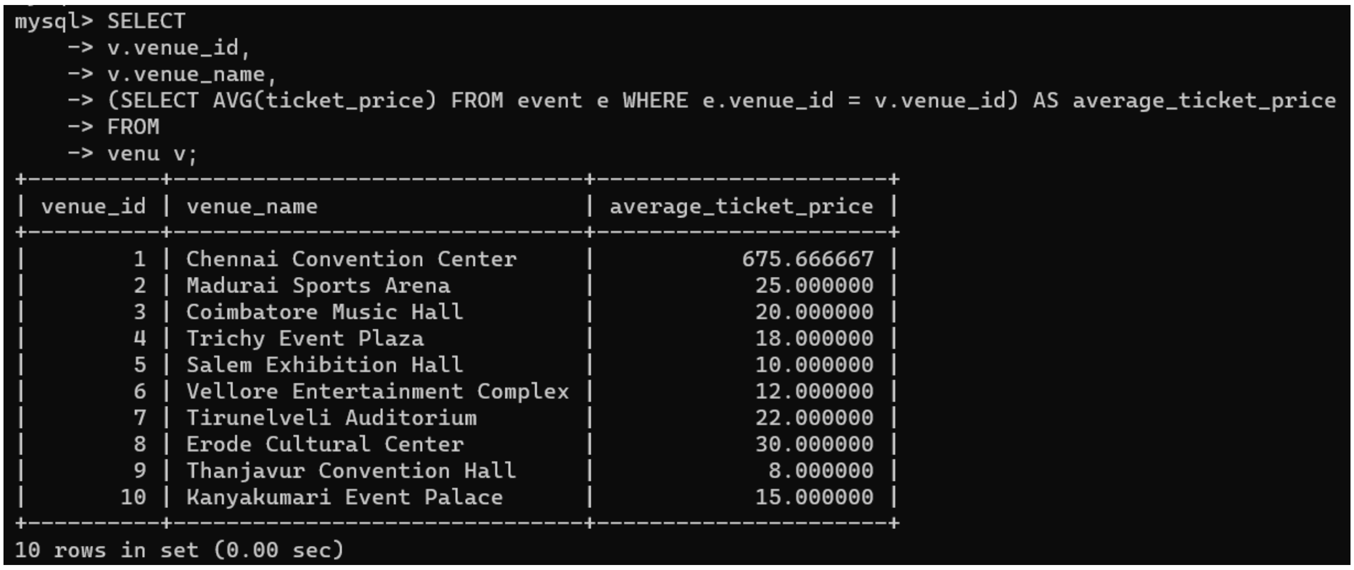
14. Write a SQL query to list Users and the Total Number of Tickets They've Purchased in the Last 30 Days.



```
mysql> SELECT
    -> c.customer_name,
    -> SUM(num_tickets) AS Ticket_Purchased,
    -> TIMESTAMPDIFF(day, MIN(booking_date), CURDATE()) AS Days_Before
    -> FROM
    -> Customer c
    -> JOIN
    -> Booking b ON b.customer_id = c.customer_id
    -> GROUP BY
    -> c.customer_name, c.customer_id
    -> HAVING
    -> Days_Before <= 30;
+-------------------+------------------+-------------+
| customer_name     | Ticket_Purchased | Days_Before |
+-------------------+------------------+-------------+
| Ananya Balaji     |                3 |          30 |
| Vishal Mohan      |                5 |          22 |
| Nithya Venkat     |                2 |          20 |
| Arjun Kumar       |                4 |          19 |
| Divya Rajan       |                6 |          18 |
| Suresh Ramalingam |                1 |          17 |
| Shreya Anand      |                3 |          16 |
+-------------------+------------------+-------------+
7 rows in set (0.00 sec)
```

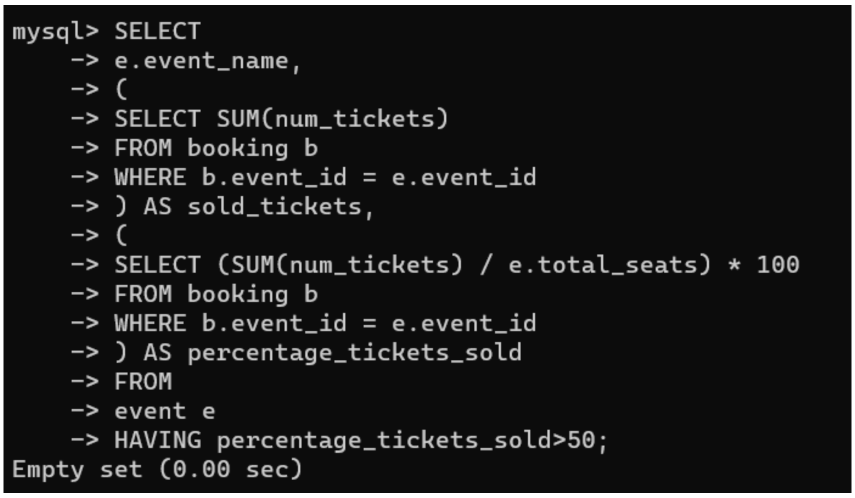
Tasks 4: Subquery and its types

1. Calculate the Average Ticket Price for Events in Each Venue Using a Subquery.



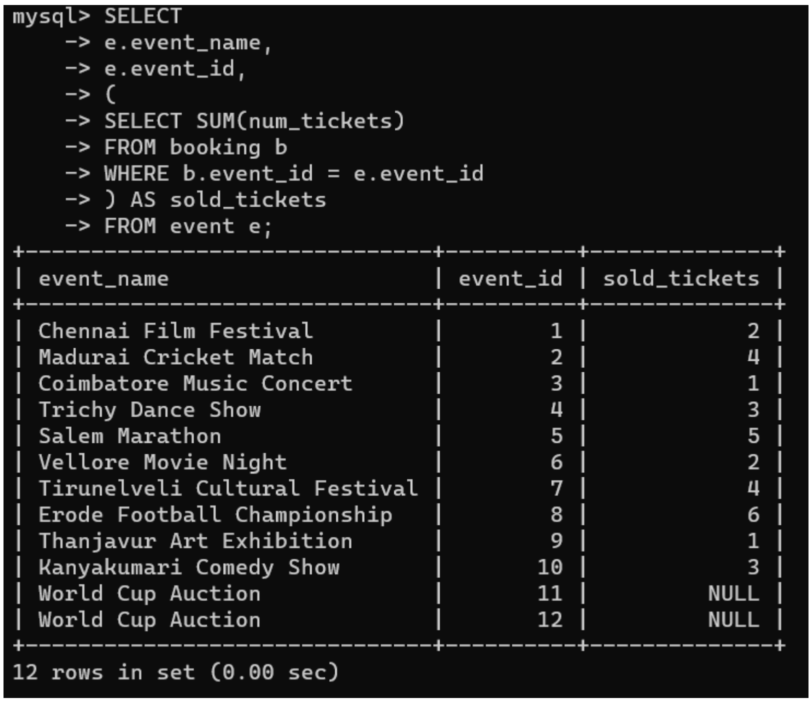
```
mysql> SELECT
    -> v.venue_id,
    -> v.venue_name,
    -> (SELECT AVG(ticket_price) FROM event e WHERE e.venue_id = v.venue_id) AS average_ticket_price
    -> FROM
    -> venu v;
+----------+-------------------------------+----------------------+
| venue_id | venue_name                    | average_ticket_price |
+----------+-------------------------------+----------------------+
|        1 | Chennai Convention Center     |           675.666667 |
|        2 | Madurai Sports Arena          |            25.000000 |
|        3 | Coimbatore Music Hall         |            20.000000 |
|        4 | Trichy Event Plaza            |            18.000000 |
|        5 | Salem Exhibition Hall         |            10.000000 |
|        6 | Vellore Entertainment Complex |            12.000000 |
|        7 | Tirunelveli Auditorium        |            22.000000 |
|        8 | Erode Cultural Center         |            30.000000 |
|        9 | Thanjavur Convention Hall     |             8.000000 |
|       10 | Kanyakumari Event Palace      |            15.000000 |
+----------+-------------------------------+----------------------+
10 rows in set (0.00 sec)
```

2. Find Events with More Than 50% of Tickets Sold using subquery.



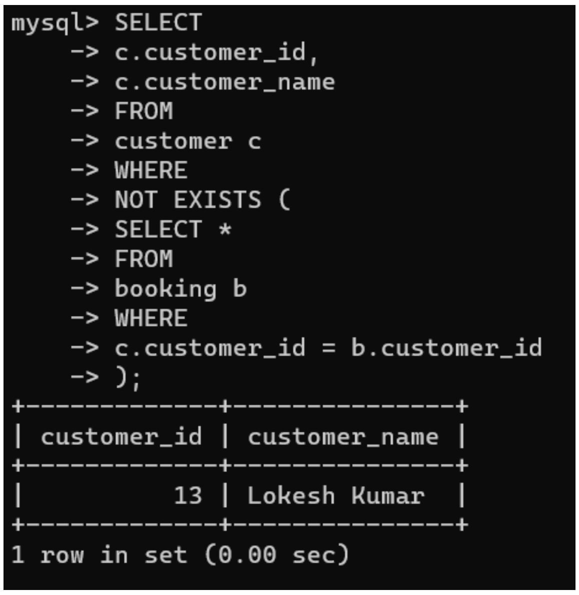
```
mysql> SELECT
    -> e.event_name,
    -> (
    -> SELECT SUM(num_tickets)
    -> FROM booking b
    -> WHERE b.event_id = e.event_id
    -> ) AS sold_tickets,
    -> (
    -> SELECT (SUM(num_tickets) / e.total_seats) * 100
    -> FROM booking b
    -> WHERE b.event_id = e.event_id
    -> ) AS percentage_tickets_sold
    -> FROM
    -> event e
    -> HAVING percentage_tickets_sold>50;
Empty set (0.00 sec)
```

3. Calculate the Total Number of Tickets Sold for Each Event.



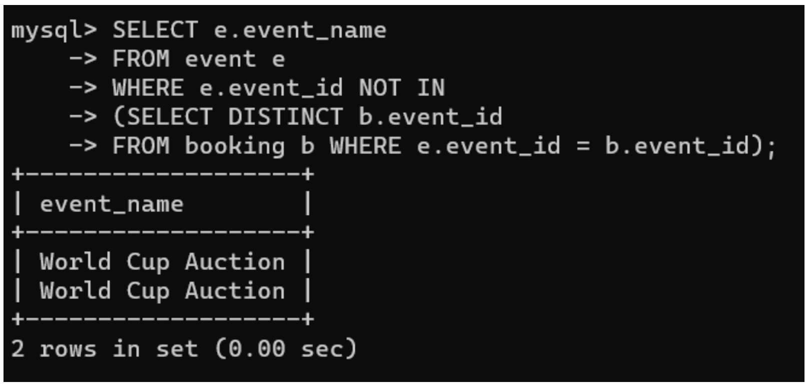
```
mysql> SELECT
    -> e.event_name,
    -> e.event_id,
    -> (
    -> SELECT SUM(num_tickets)
    -> FROM booking b
    -> WHERE b.event_id = e.event_id
    -> ) AS sold_tickets
    -> FROM event e;
+------------------------------+----------+--------------+
| event_name                   | event_id | sold_tickets |
+------------------------------+----------+--------------+
| Chennai Film Festival        |        1 |            2 |
| Madurai Cricket Match        |        2 |            4 |
| Coimbatore Music Concert     |        3 |            1 |
| Trichy Dance Show            |        4 |            3 |
| Salem Marathon               |        5 |            5 |
| Vellore Movie Night          |        6 |            2 |
| Tirunelveli Cultural Festival |       7 |            4 |
| Erode Football Championship  |        8 |            6 |
| Thanjavur Art Exhibition     |        9 |            1 |
| Kanyakumari Comedy Show      |       10 |            3 |
| World Cup Auction            |       11 |         NULL |
| World Cup Auction            |       12 |         NULL |
+------------------------------+----------+--------------+
12 rows in set (0.00 sec)
```

4. Find Users Who Have Not Booked Any Tickets Using a NOT EXISTS Subquery.



```
mysql> SELECT
    -> c.customer_id,
    -> c.customer_name
    -> FROM
    -> customer c
    -> WHERE
    -> NOT EXISTS (
    -> SELECT *
    -> FROM
    -> booking b
    -> WHERE
    -> c.customer_id = b.customer_id
    -> );
+-------------+---------------+
| customer_id | customer_name |
+-------------+---------------+
|          13 | Lokesh Kumar  |
+-------------+---------------+
1 row in set (0.00 sec)
```

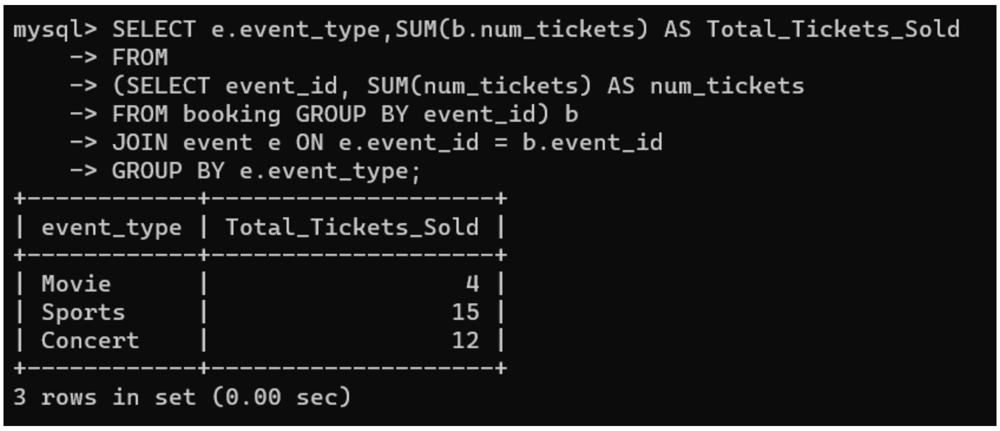
5. List Events with No Ticket Sales Using a NOT IN Subquery.



```
mysql> SELECT e.event_name
    -> FROM event e
    -> WHERE e.event_id NOT IN
    -> (SELECT DISTINCT b.event_id
    -> FROM booking b WHERE e.event_id = b.event_id);
+-------------------+
| event_name        |
+-------------------+
| World Cup Auction |
| World Cup Auction |
+-------------------+
2 rows in set (0.00 sec)
```

6. Calculate the Total Number of Tickets Sold for Each Event Type Using a Subquery in the FROM

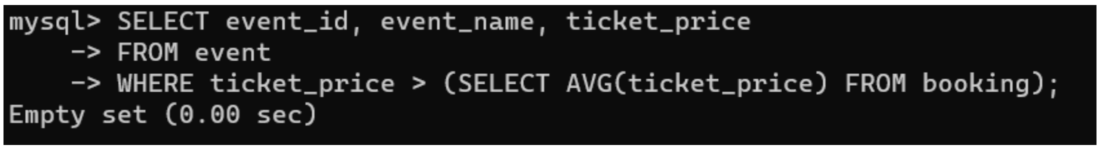
Clause.



```
mysql> SELECT e.event_type,SUM(b.num_tickets) AS Total_Tickets_Sold
    -> FROM
    -> (SELECT event_id, SUM(num_tickets) AS num_tickets
    -> FROM booking GROUP BY event_id) b
    -> JOIN event e ON e.event_id = b.event_id
    -> GROUP BY e.event_type;
+------------+--------------------+
| event_type | Total_Tickets_Sold |
+------------+--------------------+
| Movie      |                  4 |
| Sports     |                 15 |
| Concert    |                 12 |
+------------+--------------------+
3 rows in set (0.00 sec)
```

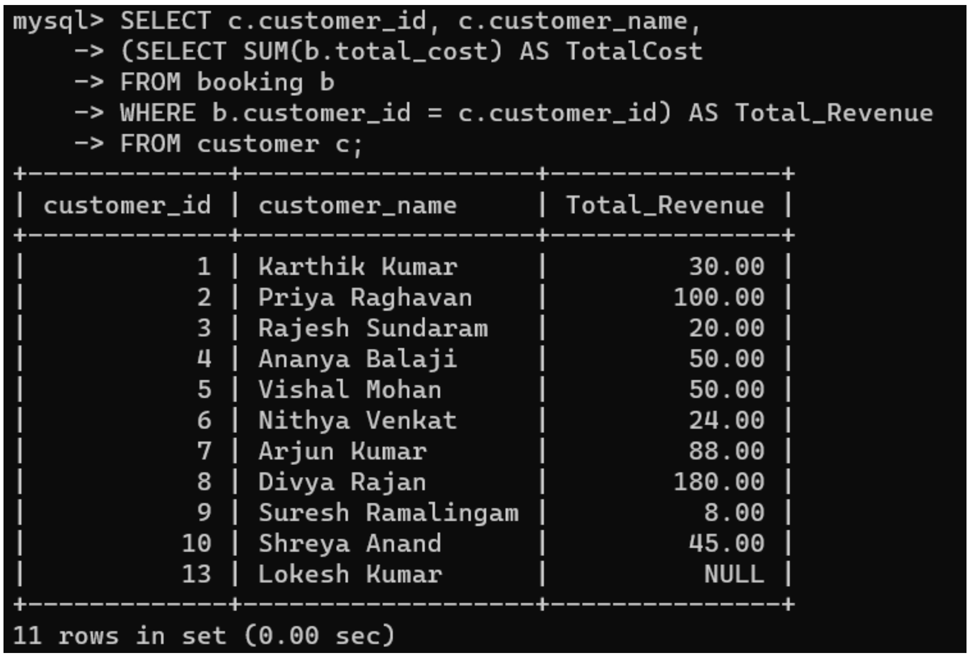
7. Find Events with Ticket Prices Higher Than the Average Ticket Price Using a Subquery in the

WHERE Clause.



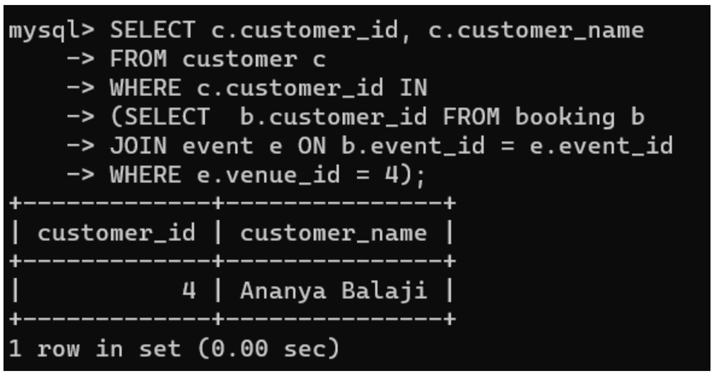
```
mysql> SELECT event_id, event_name, ticket_price
    -> FROM event
    -> WHERE ticket_price > (SELECT AVG(ticket_price) FROM booking);
Empty set (0.00 sec)
```

8. Calculate the Total Revenue Generated by Events for Each User Using a Correlated Subquery.



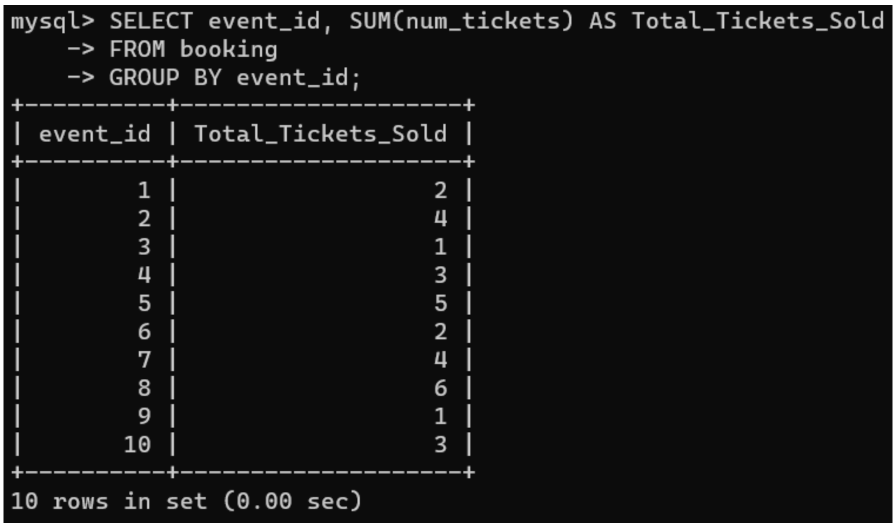
```
mysql> SELECT c.customer_id, c.customer_name,
    -> (SELECT SUM(b.total_cost) AS TotalCost
    -> FROM booking b
    -> WHERE b.customer_id = c.customer_id) AS Total_Revenue
    -> FROM customer c;
+-------------+-------------------+---------------+
| customer_id | customer_name     | Total_Revenue |
+-------------+-------------------+---------------+
|           1 | Karthik Kumar     |         30.00 |
|           2 | Priya Raghavan    |        100.00 |
|           3 | Rajesh Sundaram   |         20.00 |
|           4 | Ananya Balaji     |         50.00 |
|           5 | Vishal Mohan      |         50.00 |
|           6 | Nithya Venkat     |         24.00 |
|           7 | Arjun Kumar       |         88.00 |
|           8 | Divya Rajan       |        180.00 |
|           9 | Suresh Ramalingam |          8.00 |
|          10 | Shreya Anand      |         45.00 |
|          13 | Lokesh Kumar      |          NULL |
+-------------+-------------------+---------------+
11 rows in set (0.00 sec)
```

9. List Users Who Have Booked Tickets for Events in a Given Venue Using a Subquery in the WHERE Clause.



```
mysql> SELECT c.customer_id, c.customer_name
    -> FROM customer c
    -> WHERE c.customer_id IN
    -> (SELECT  b.customer_id FROM booking b
    -> JOIN event e ON b.event_id = e.event_id
    -> WHERE e.venue_id = 4);
+-------------+---------------+
| customer_id | customer_name |
+-------------+---------------+
|           4 | Ananya Balaji |
+-------------+---------------+
1 row in set (0.00 sec)
```

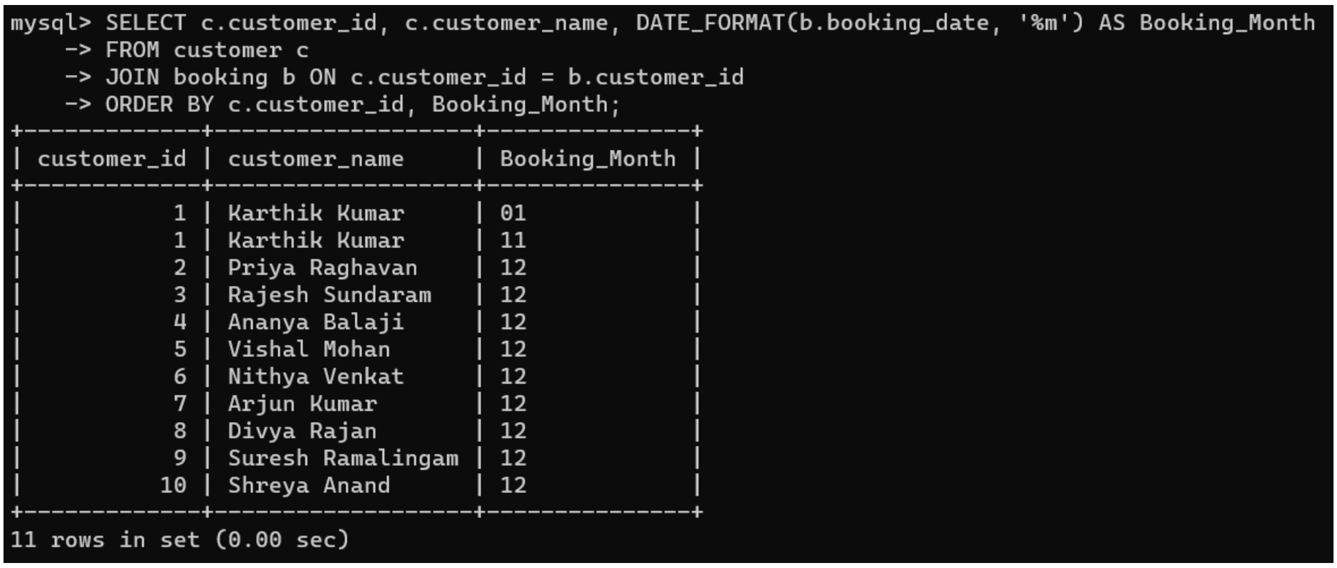
10. Calculate the Total Number of Tickets Sold for Each Event Category Using a Subquery with GROUP BY.



```
mysql> SELECT event_id, SUM(num_tickets) AS Total_Tickets_Sold
    -> FROM booking
    -> GROUP BY event_id;
+----------+--------------------+
| event_id | Total_Tickets_Sold |
+----------+--------------------+
|        1 |                  2 |
|        2 |                  4 |
|        3 |                  1 |
|        4 |                  3 |
|        5 |                  5 |
|        6 |                  2 |
|        7 |                  4 |
|        8 |                  6 |
|        9 |                  1 |
|       10 |                  3 |
+----------+--------------------+
10 rows in set (0.00 sec)
```

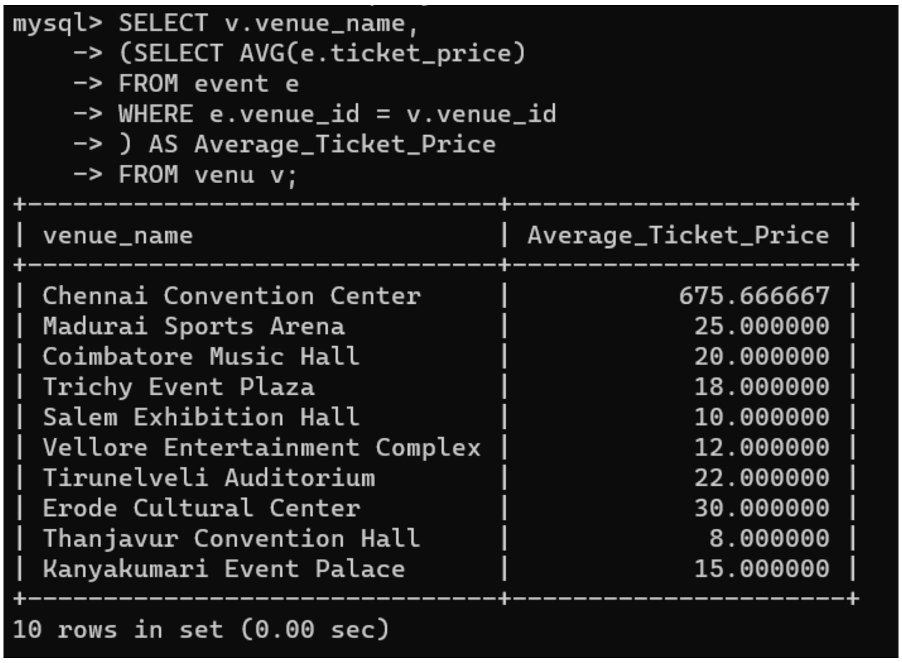
11. Find Users Who Have Booked Tickets for Events in each Month Using a Subquery with

DATE_FORMAT.



```
mysql> SELECT c.customer_id, c.customer_name, DATE_FORMAT(b.booking_date, '%m') AS Booking_Month
    -> FROM customer c
    -> JOIN booking b ON c.customer_id = b.customer_id
    -> ORDER BY c.customer_id, Booking_Month;
+-------------+-------------------+---------------+
| customer_id | customer_name     | Booking_Month |
+-------------+-------------------+---------------+
|           1 | Karthik Kumar     | 01            |
|           1 | Karthik Kumar     | 11            |
|           2 | Priya Raghavan    | 12            |
|           3 | Rajesh Sundaram   | 12            |
|           4 | Ananya Balaji     | 12            |
|           5 | Vishal Mohan      | 12            |
|           6 | Nithya Venkat     | 12            |
|           7 | Arjun Kumar       | 12            |
|           8 | Divya Rajan       | 12            |
|           9 | Suresh Ramalingam | 12            |
|          10 | Shreya Anand      | 12            |
+-------------+-------------------+---------------+
11 rows in set (0.00 sec)
```

12. Calculate the Average Ticket Price for Events in Each Venue Using a Subquery



```
mysql> SELECT v.venue_name,
    -> (SELECT AVG(e.ticket_price)
    -> FROM event e
    -> WHERE e.venue_id = v.venue_id
    -> ) AS Average_Ticket_Price
    -> FROM venu v;
+-------------------------------+----------------------+
| venue_name                    | Average_Ticket_Price |
+-------------------------------+----------------------+
| Chennai Convention Center     |           675.666667 |
| Madurai Sports Arena          |            25.000000 |
| Coimbatore Music Hall         |            20.000000 |
| Trichy Event Plaza            |            18.000000 |
| Salem Exhibition Hall         |            10.000000 |
| Vellore Entertainment Complex |            12.000000 |
| Tirunelveli Auditorium        |            22.000000 |
| Erode Cultural Center         |            30.000000 |
| Thanjavur Convention Hall     |             8.000000 |
| Kanyakumari Event Palace      |            15.000000 |
+-------------------------------+----------------------+
10 rows in set (0.00 sec)
```