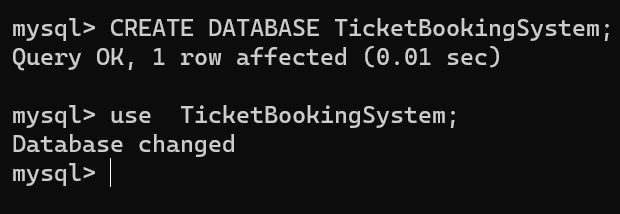
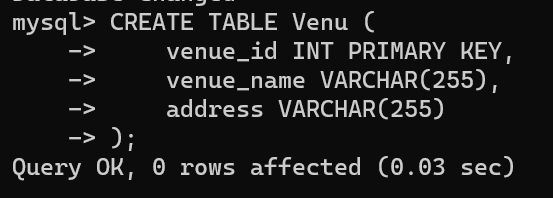
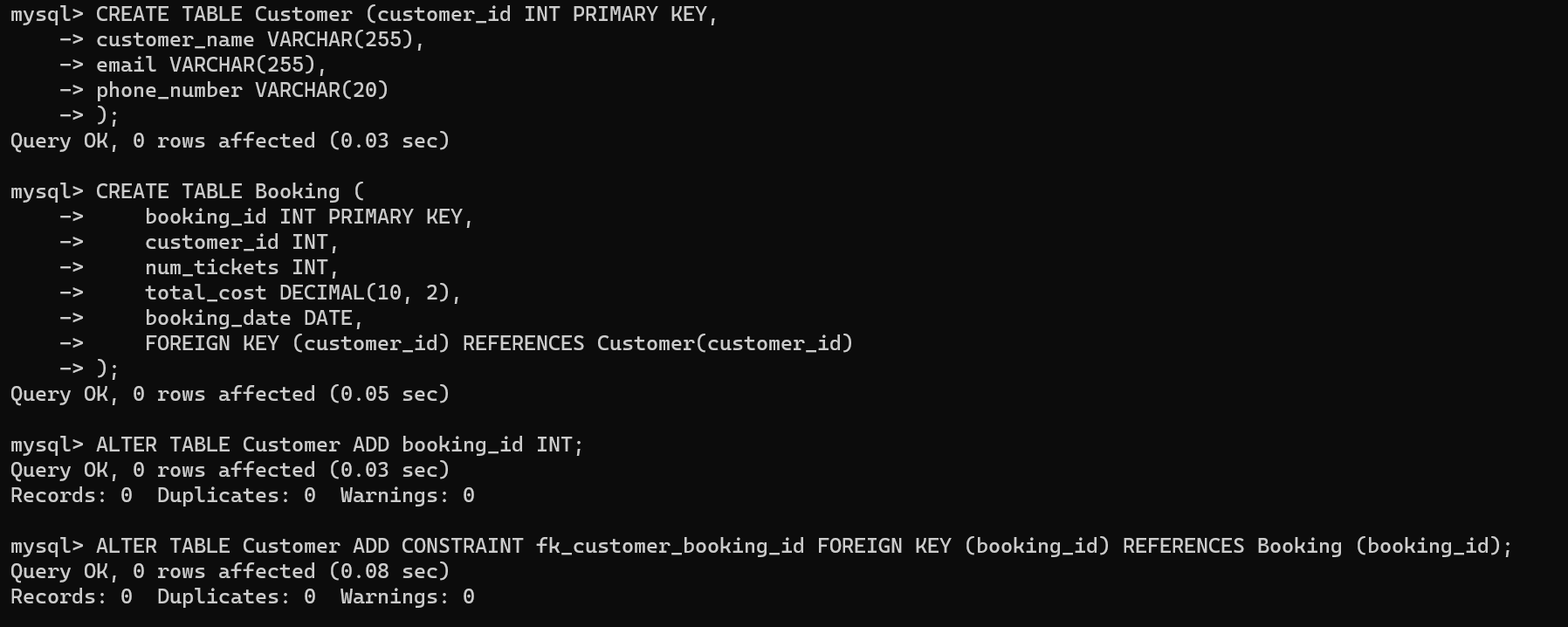
**Tasks 1: Database Design:**

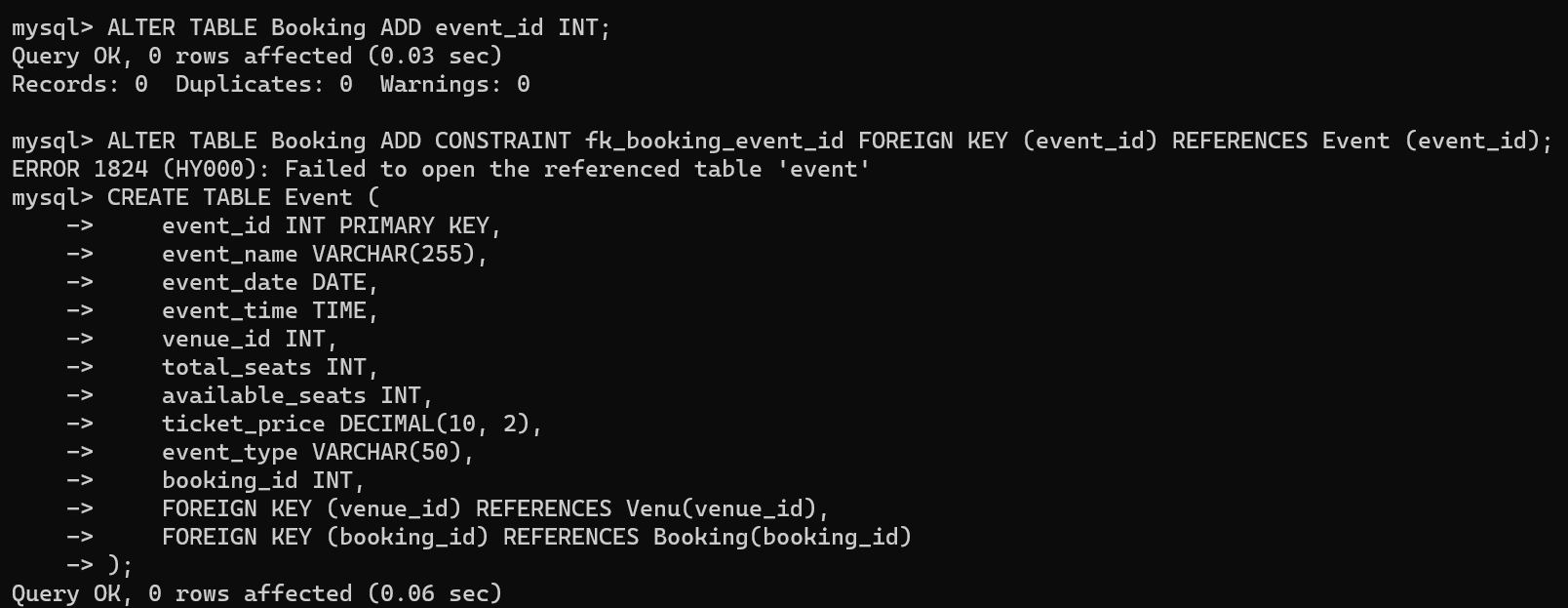
**1. Create the database named "TicketBookingSystem"**

****

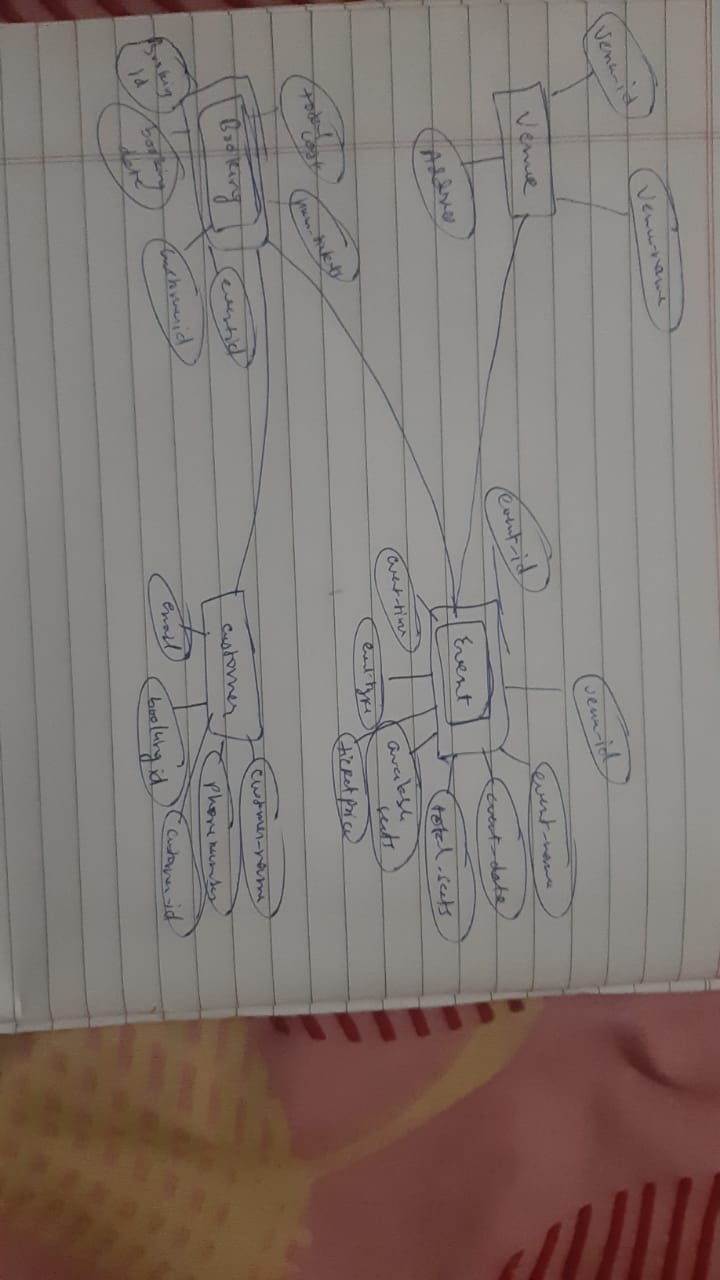
**2. Write SQL scripts to create the mentioned tables with appropriate data types, constraints, and relationships. • Venu • Event • Customers • Booking**

****

****

****

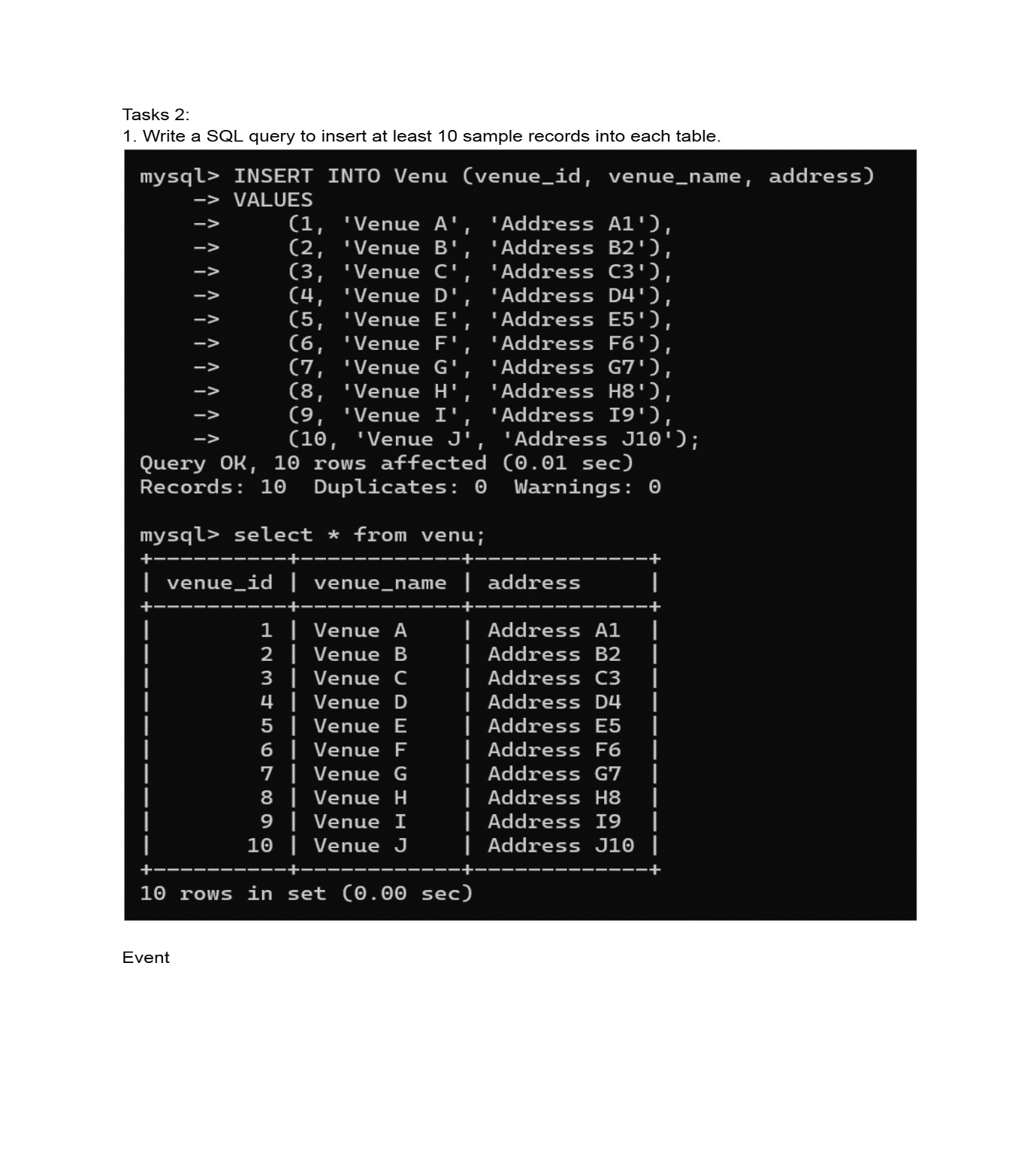
**3. Create an ERD (Entity Relationship Diagram) for the database.**

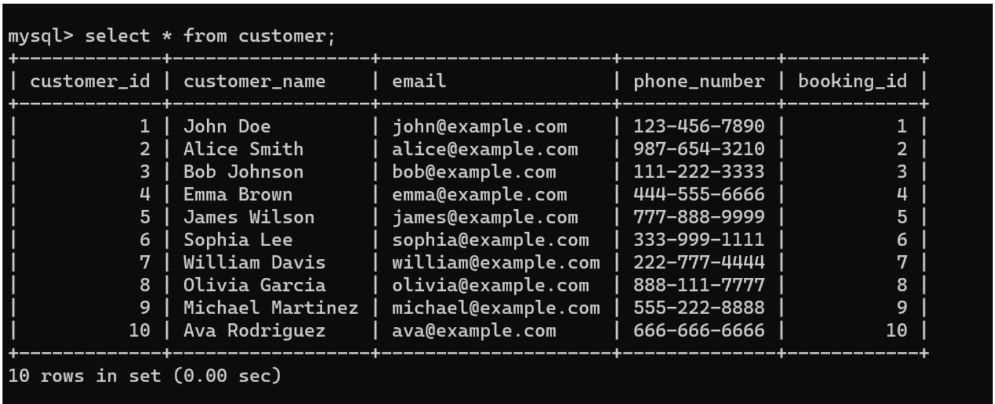


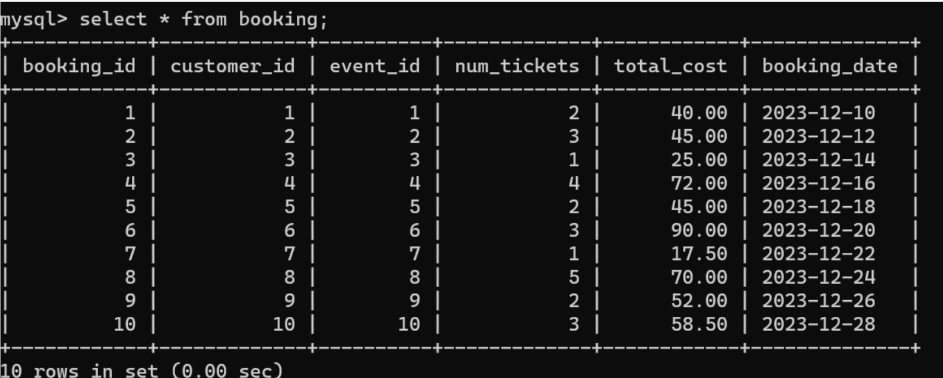
**4. Create appropriate Primary Key and Foreign Key constraints for referential integrity.**

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

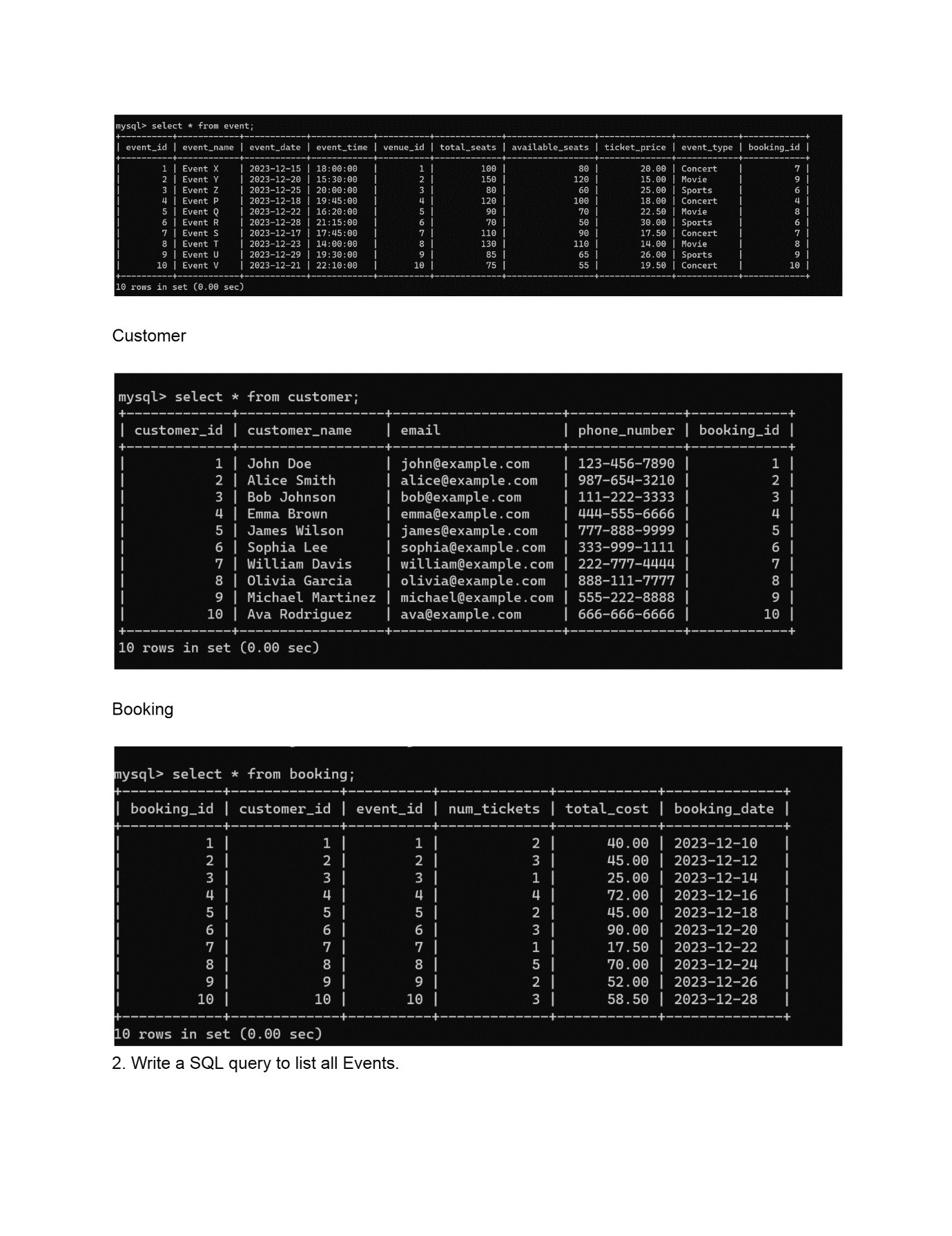
**1. Write a SQL query to insert at least 10 sample records into each table.**



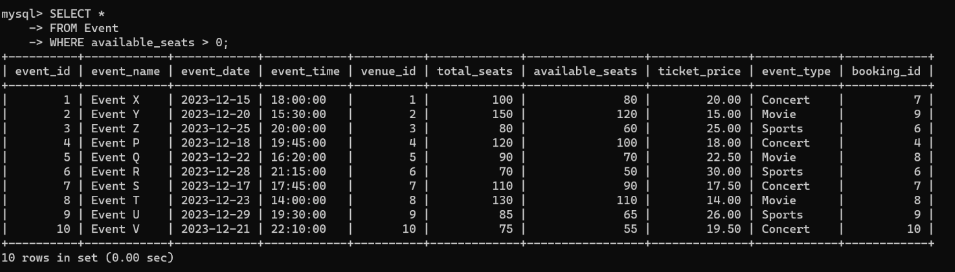
****

****

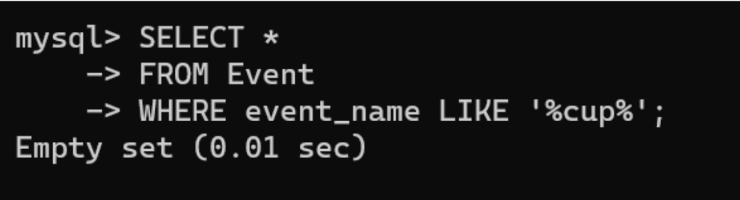
**2. Write a SQL query to list all Events.**



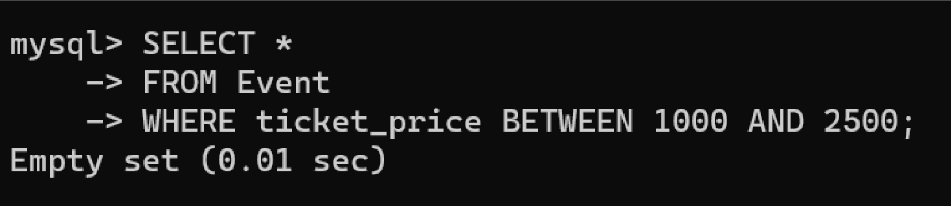
**3. Write a SQL query to select events with available tickets.**

****

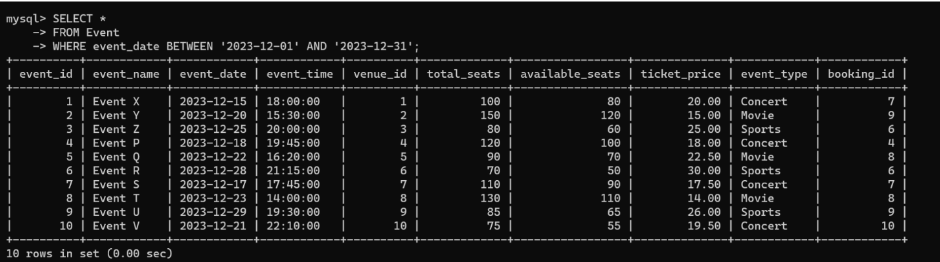
**4. Write a SQL query to select events name partial match with ‘cup’.**

****

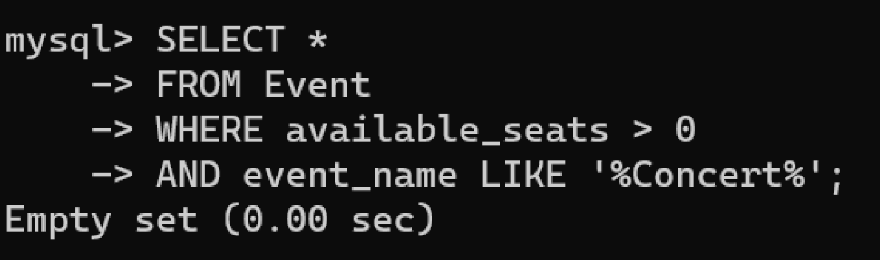
**5. Write a SQL query to select events with ticket price range is between 1000 to 2500.**

****

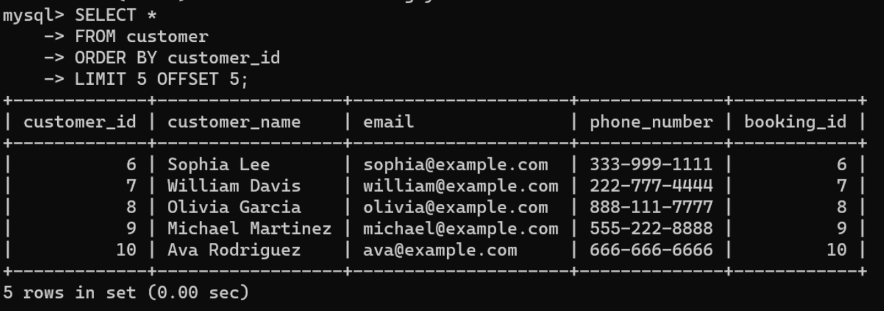
**6. Write a SQL query to retrieve events with dates falling within a specific range.**

****

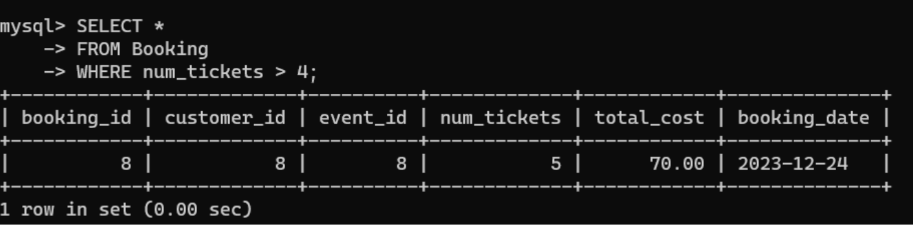
**7. Write a SQL query to retrieve events with available tickets that also have "Concert" in their name.**

****

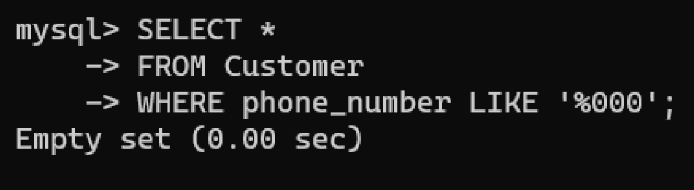
**8. Write a SQL query to retrieve users in batches of 5, starting from the 6th user.**

****

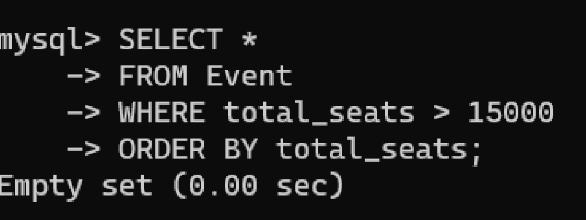
**9. Write a SQL query to retrieve bookings details contains booked no of ticket more than 4.**

****

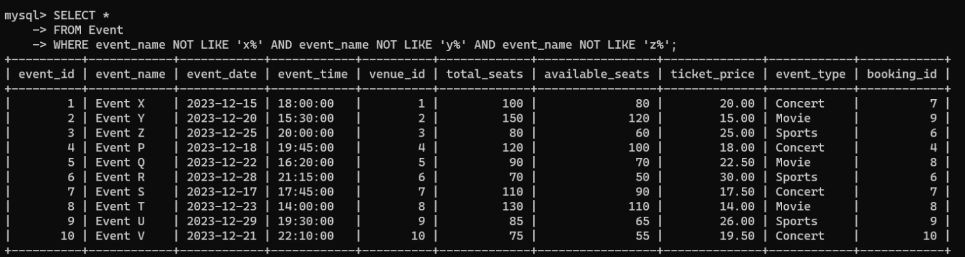
**10. Write a SQL query to retrieve customer information whose phone number end with ‘000’**

****

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

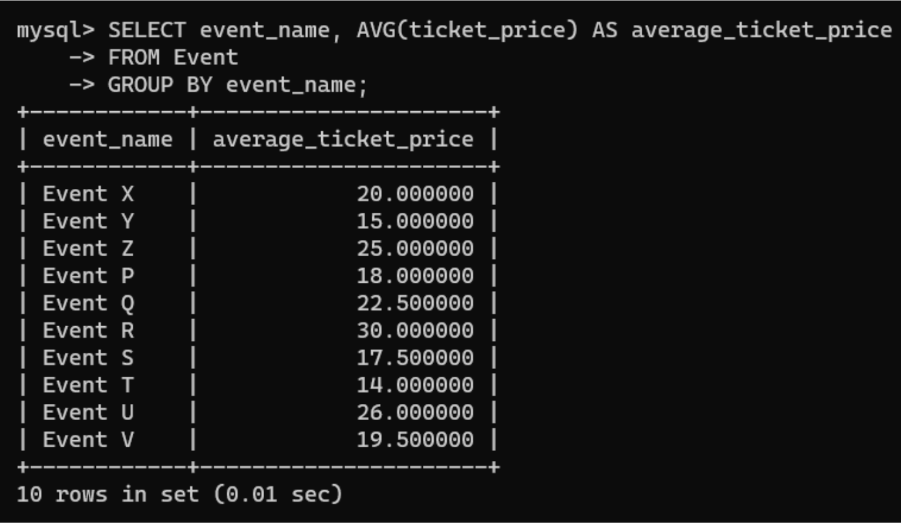
****

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

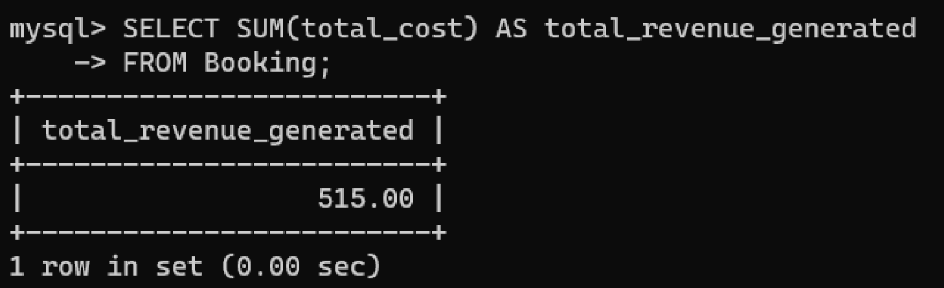
****

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

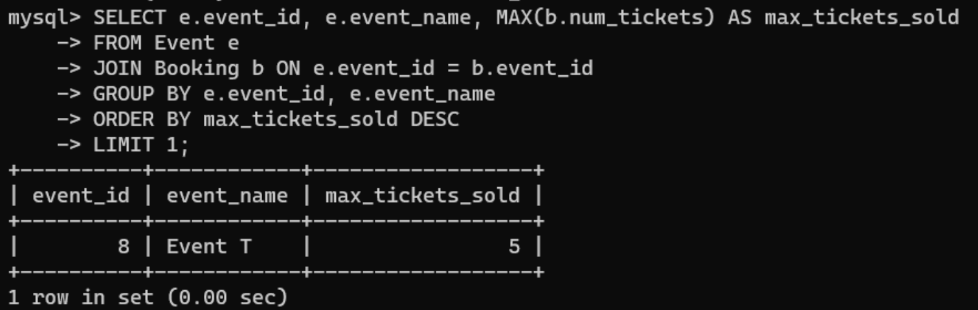
**1. Write a SQL query to List Events and Their Average Ticket Prices.**

****

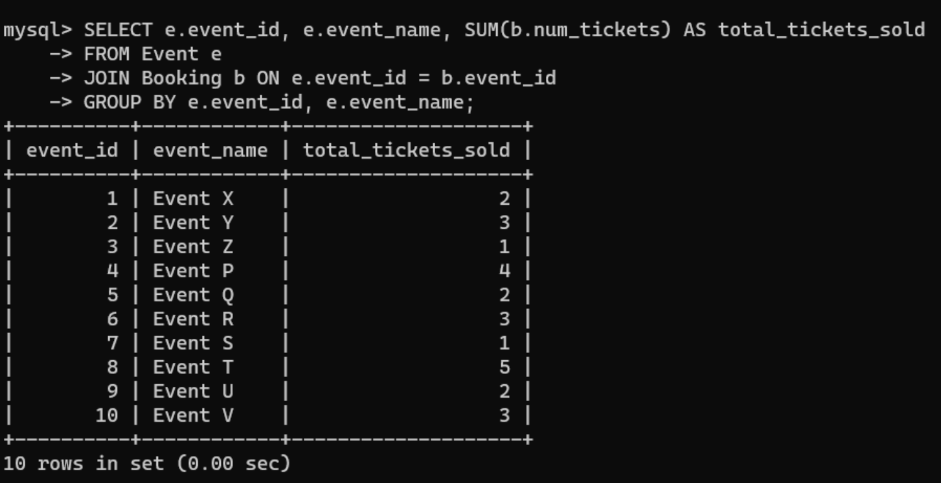
**2. Write a SQL query to Calculate the Total Revenue Generated by Events.**

****

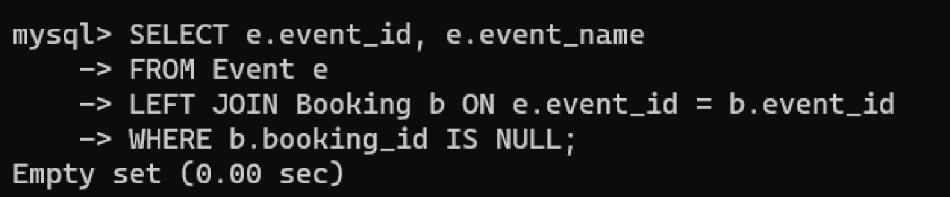
**3. Write a SQL query to find the event with the highest ticket sales.**

****

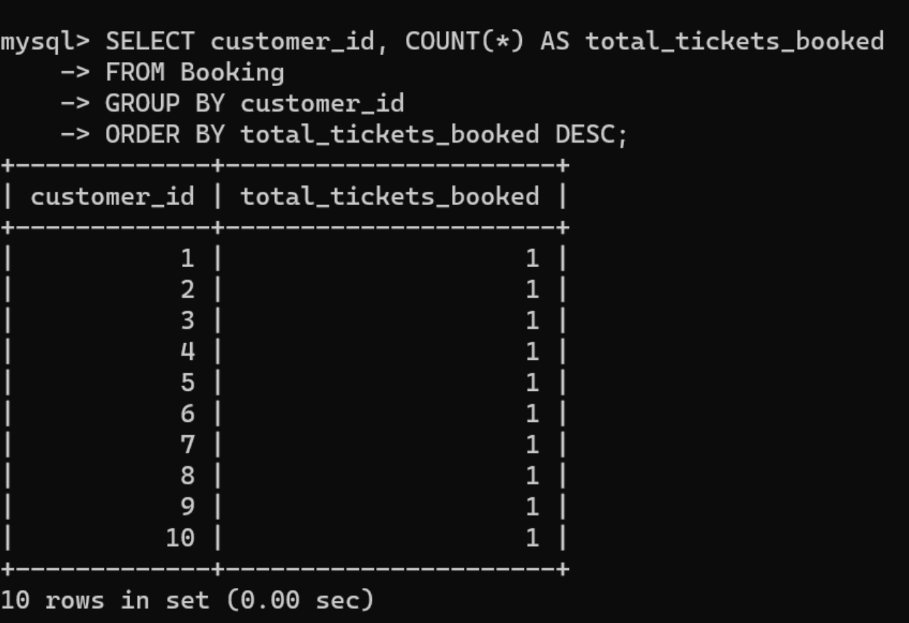
**4. Write a SQL query to Calculate the Total Number of Tickets Sold for Each Event.**

****

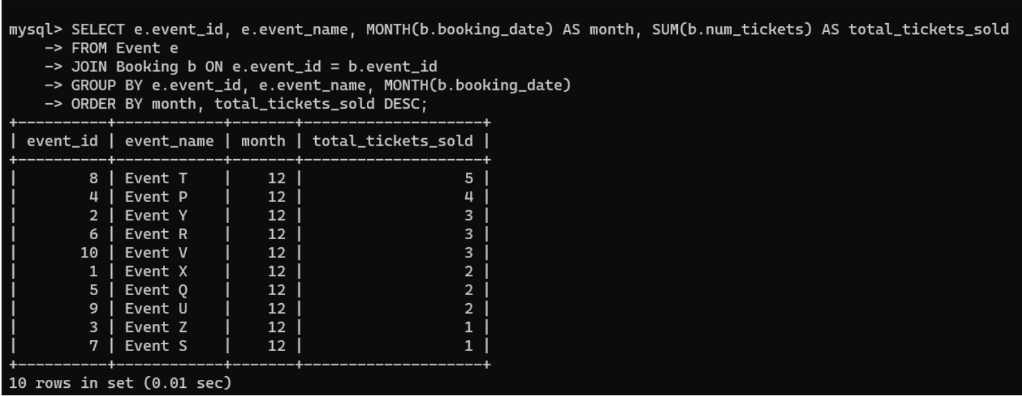
**5. Write a SQL query to Find Events with No Ticket Sales.**

****

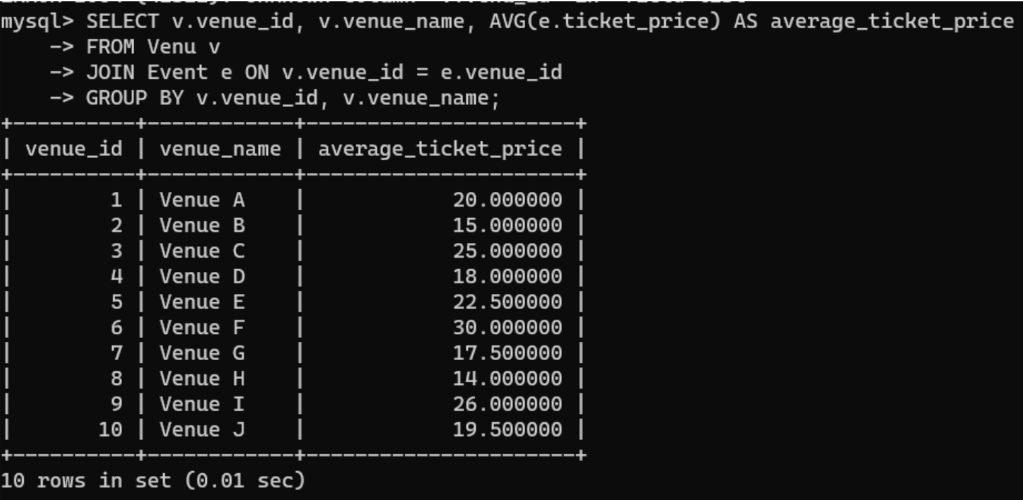
**6. Write a SQL query to Find the User Who Has Booked the Most Tickets.**

****

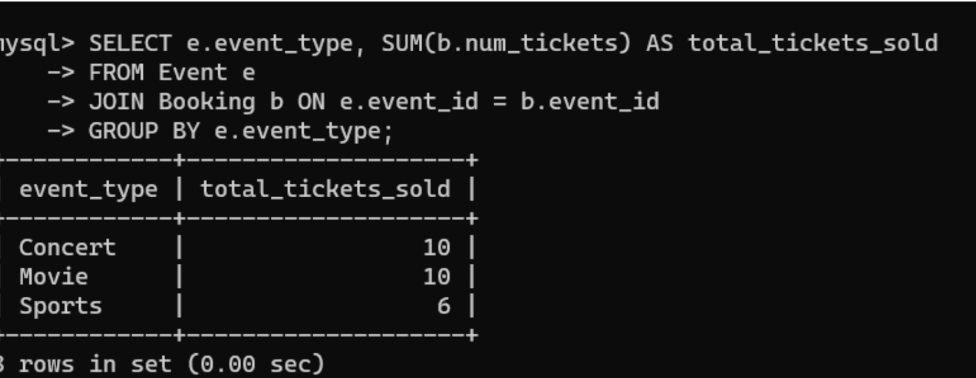
**7. Write a SQL query to List Events and the total number of tickets sold for each month.**

****

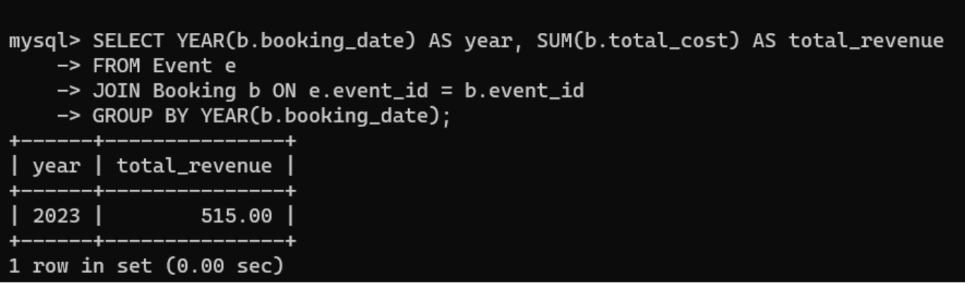
**8. Write a SQL query to calculate the average Ticket Price for Events in Each Venue.**

****

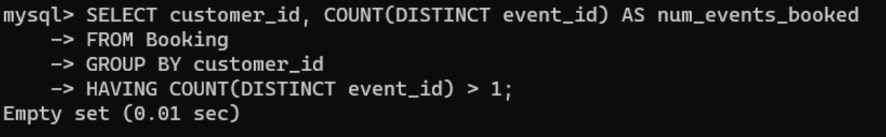
**9. Write a SQL query to calculate the total Number of Tickets Sold for Each Event Type.**

****

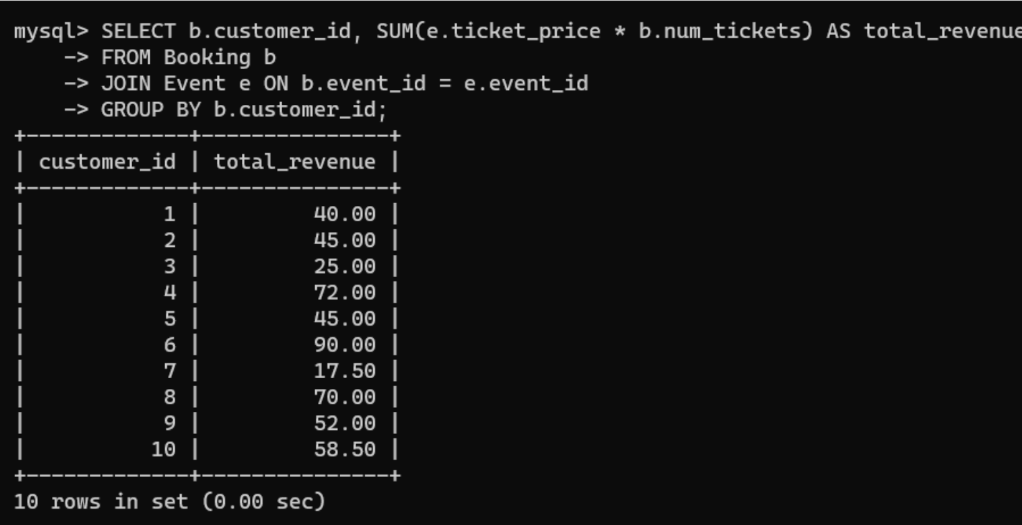
**10. Write a SQL query to calculate the total Revenue Generated by Events in Each Year.**

****

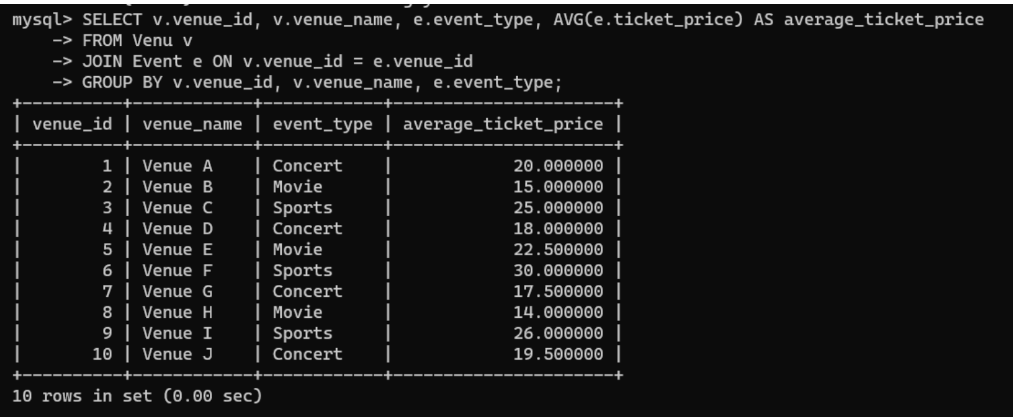
**11. Write a SQL query to list users who have booked tickets for multiple events.**

****

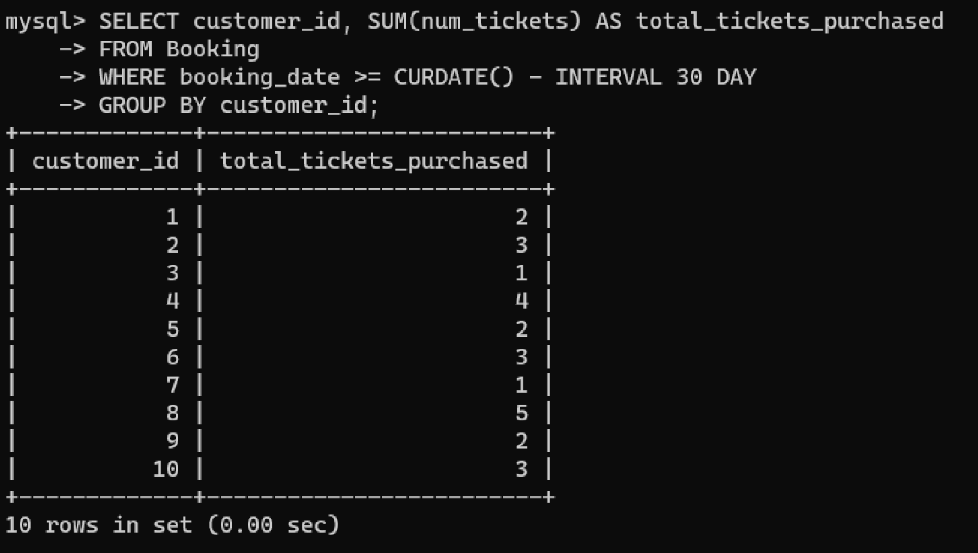
**12. Write a SQL query to calculate the Total Revenue Generated by Events for Each User.**

****

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

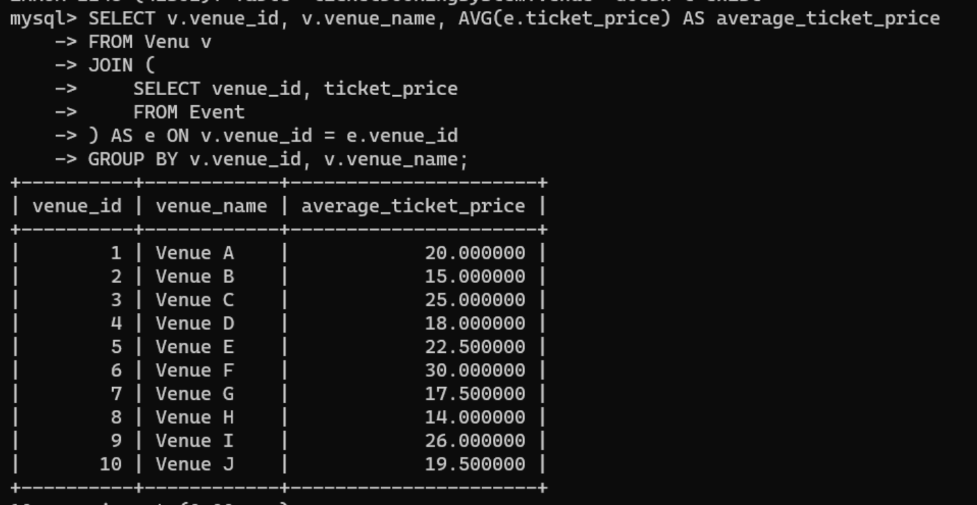
****

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

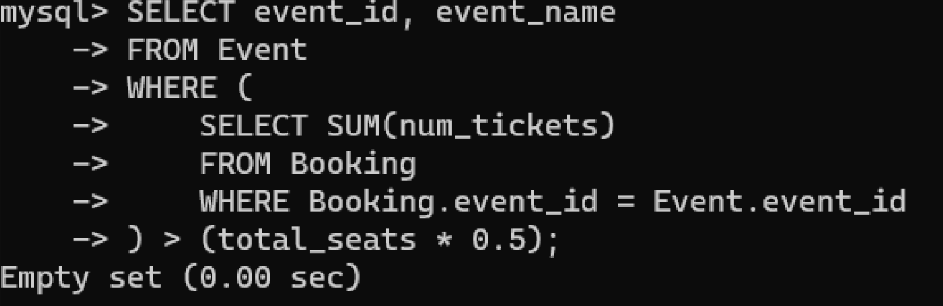
****

**Tasks 4: Subquery and its types**

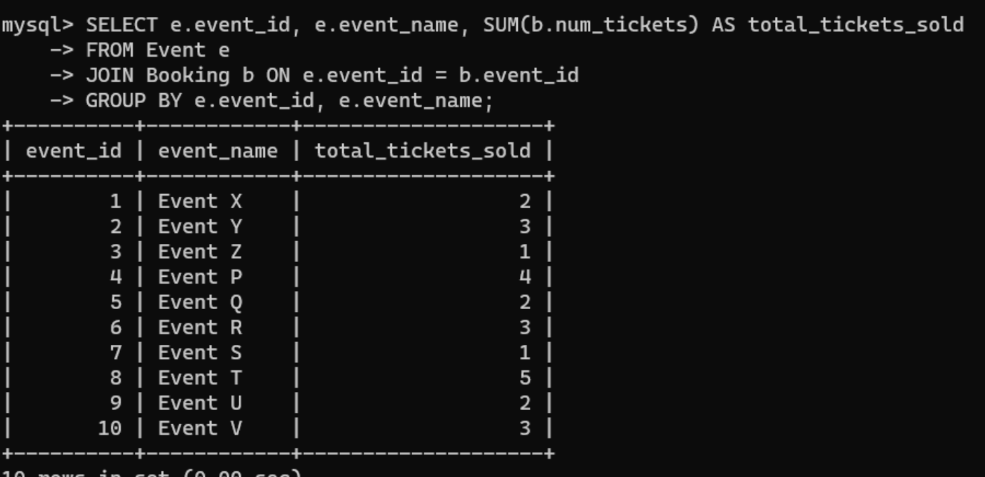
**1. Calculate the Average Ticket Price for Events in Each Venue Using a Subquery.**

****

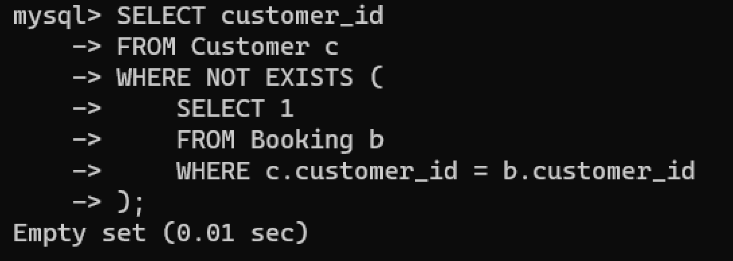
**2. Find Events with More Than 50% of Tickets Sold using subquery.**

****

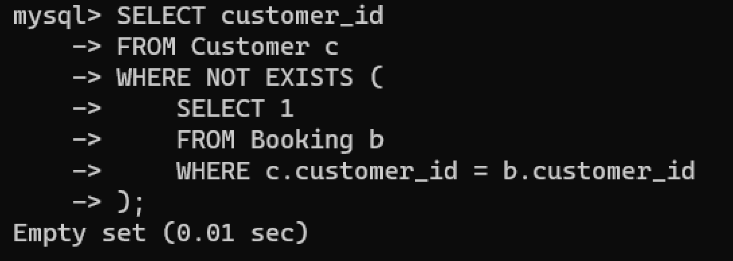
**3. Calculate the Total Number of Tickets Sold for Each Event.**

****

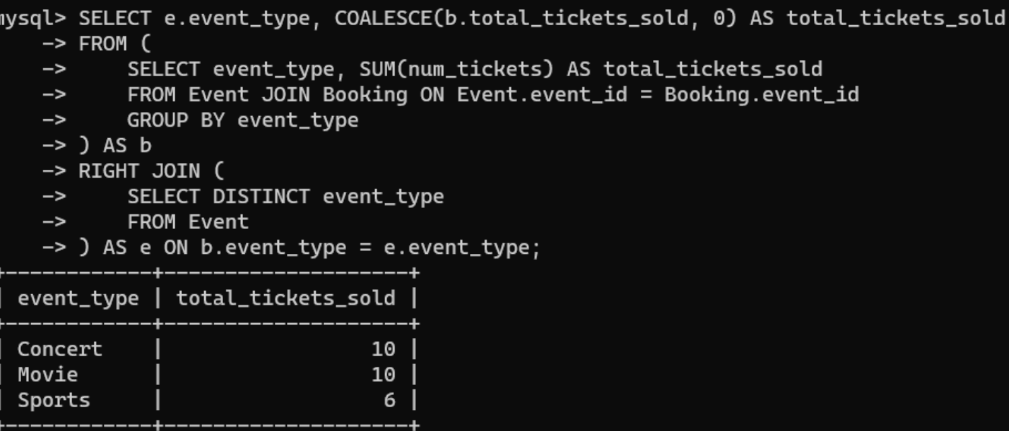
**4. Find Users Who Have Not Booked Any Tickets Using a NOT EXISTS Subquery.**

****

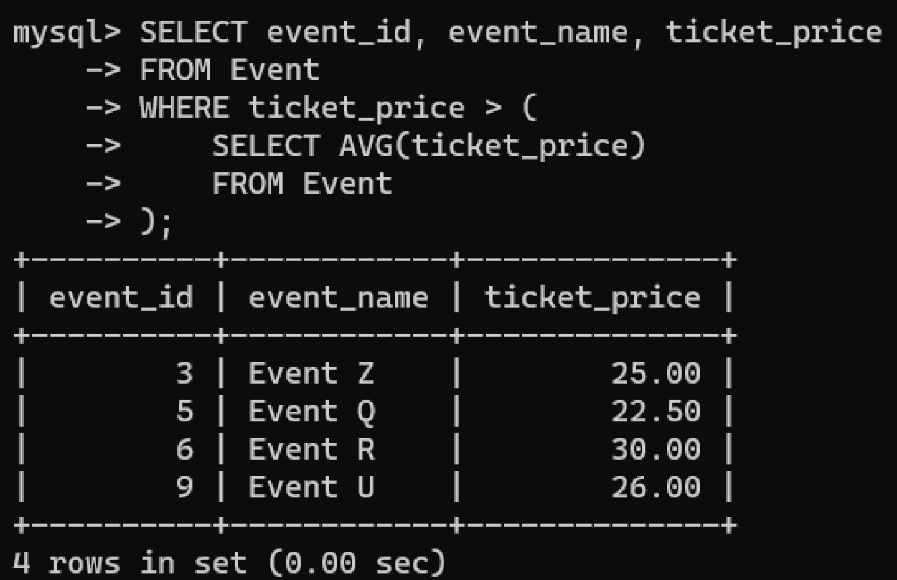
**5. List Events with No Ticket Sales Using a NOT IN Subquery.**

****

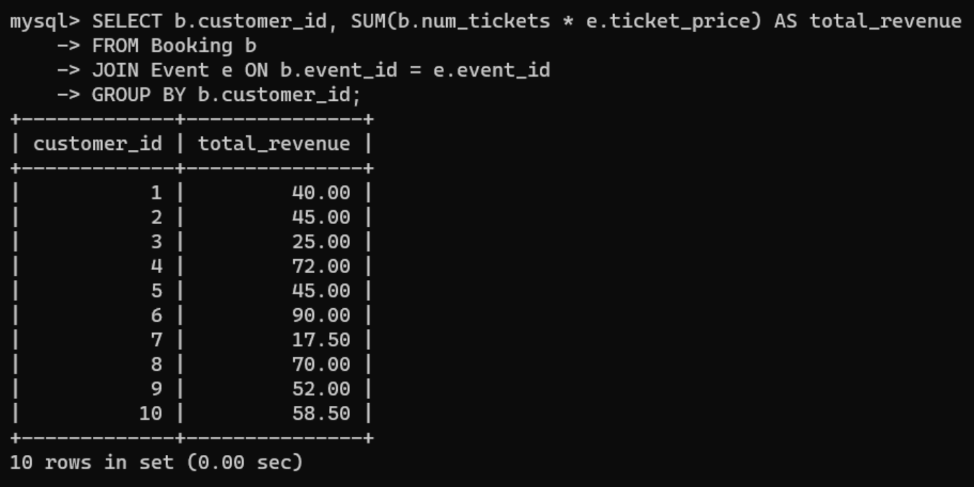
**6. Calculate the Total Number of Tickets Sold for Each Event Type Using a Subquery in the FROM Clause.**

****

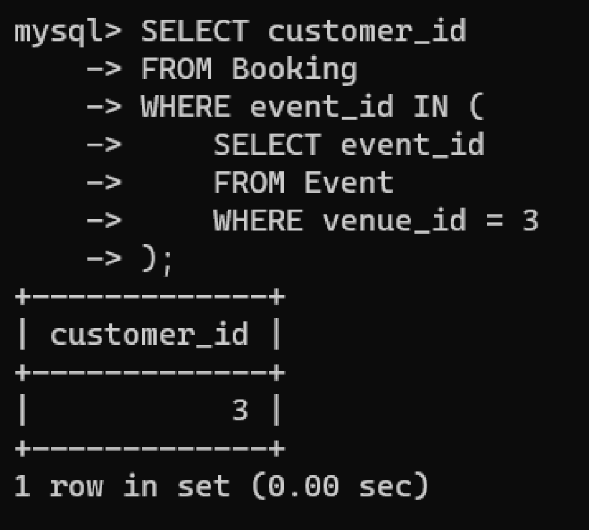
**7. Find Events with Ticket Prices Higher Than the Average Ticket Price Using a Subquery in the WHERE Clause.**

****

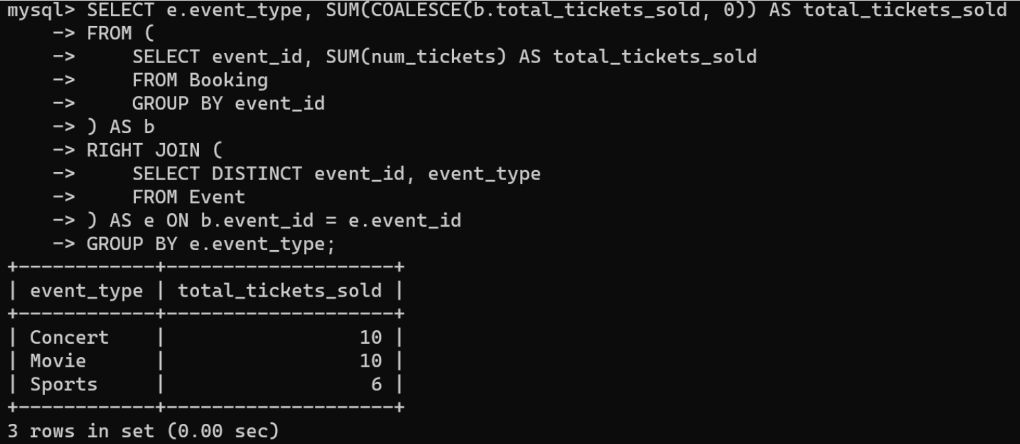
**8. Calculate the Total Revenue Generated by Events for Each User Using a Correlated Subquery.**

****

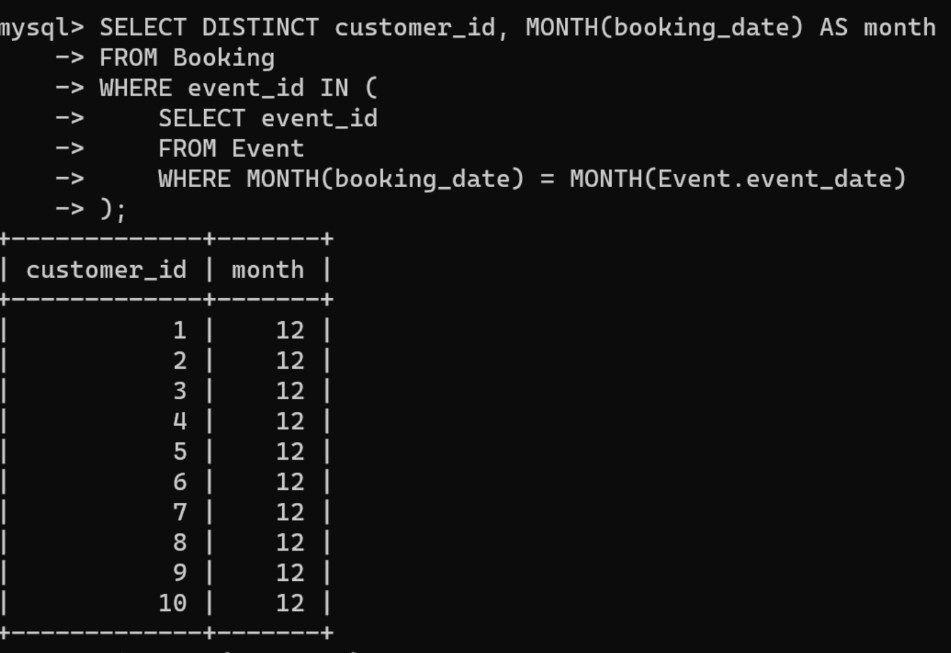
**9. List Users Who Have Booked Tickets for Events in a Given Venue Using a Subquery in the WHERE Clause.**

****

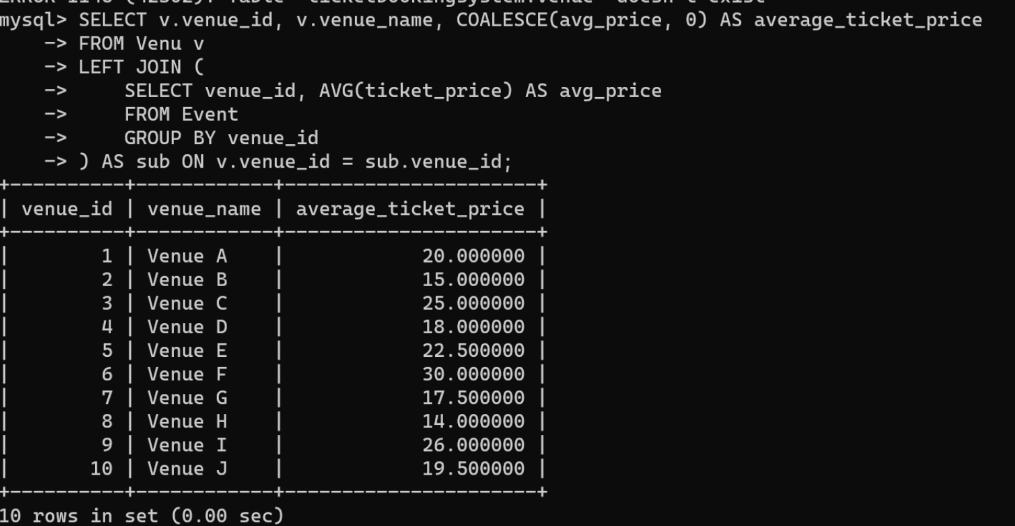
**10. Calculate the Total Number of Tickets Sold for Each Event Category Using a Subquery with GROUP BY.**

****

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

****

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

****