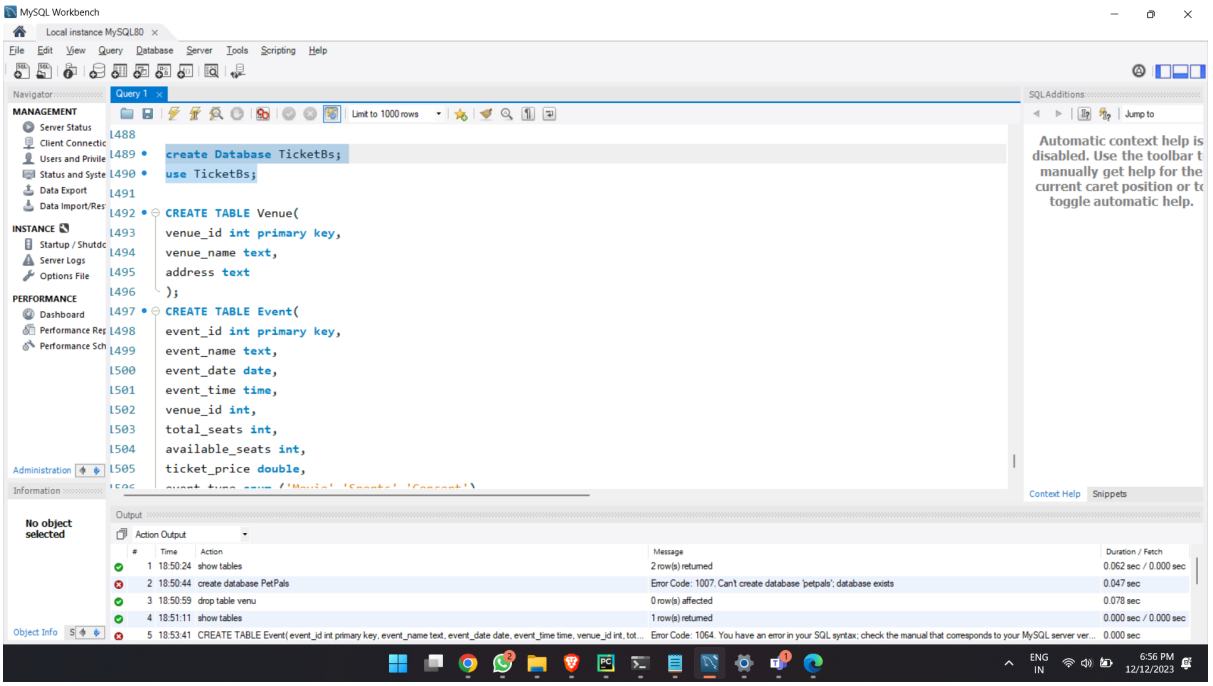
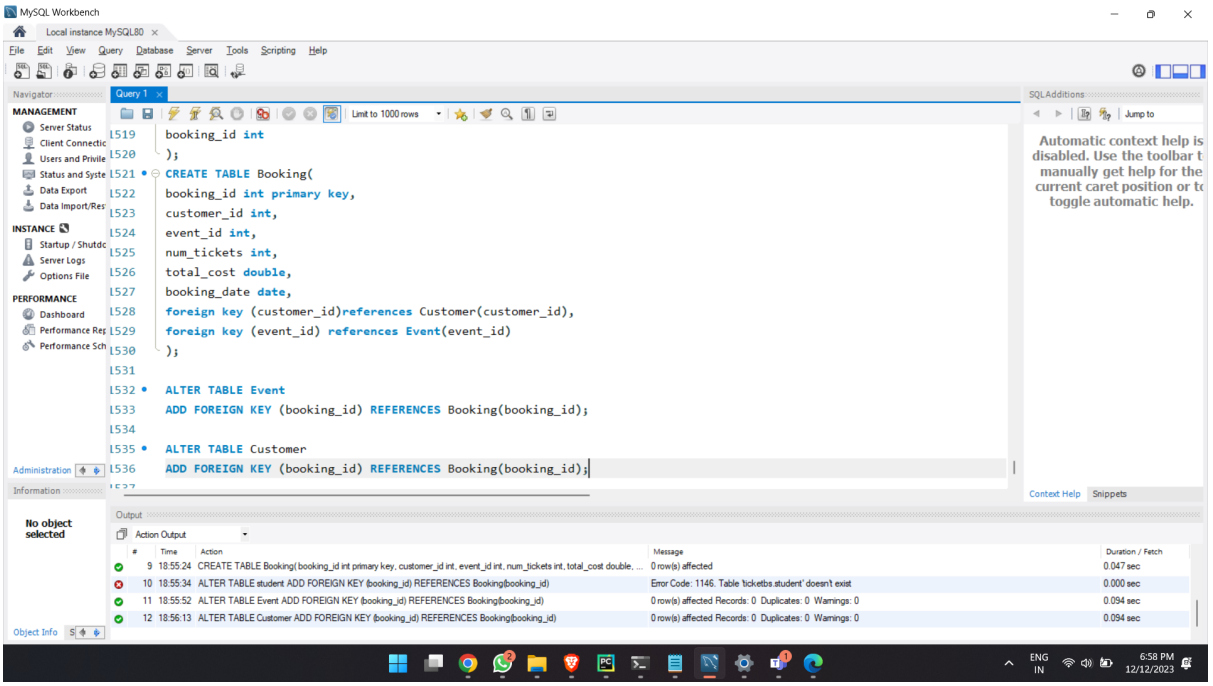
Tasks 1: Database Design:

Q1. Create the database named "TicketBookingSystem"

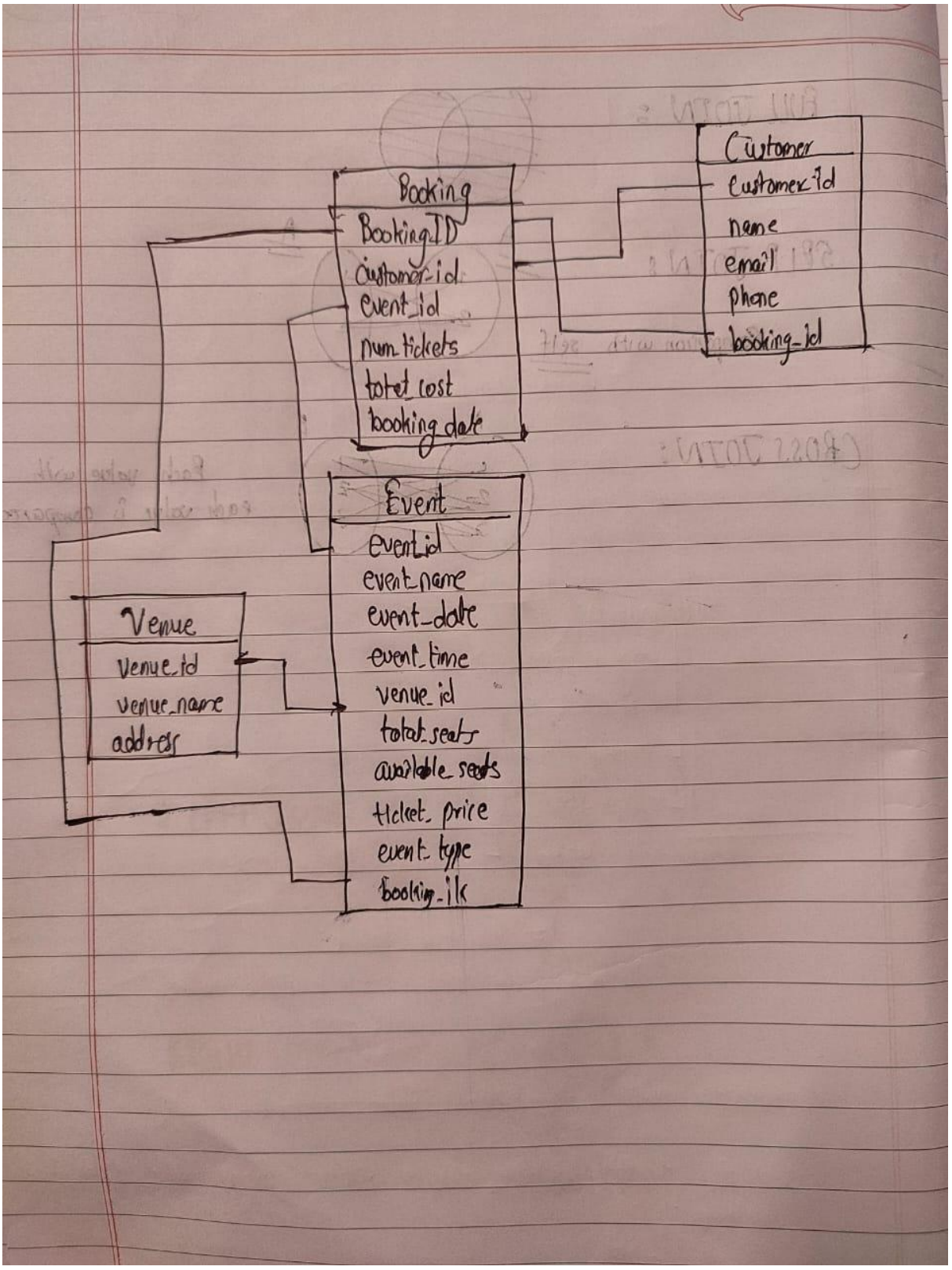


Q2. Write SQL scripts to create the mentioned tables with appropriate data types, constraints, and relationships.

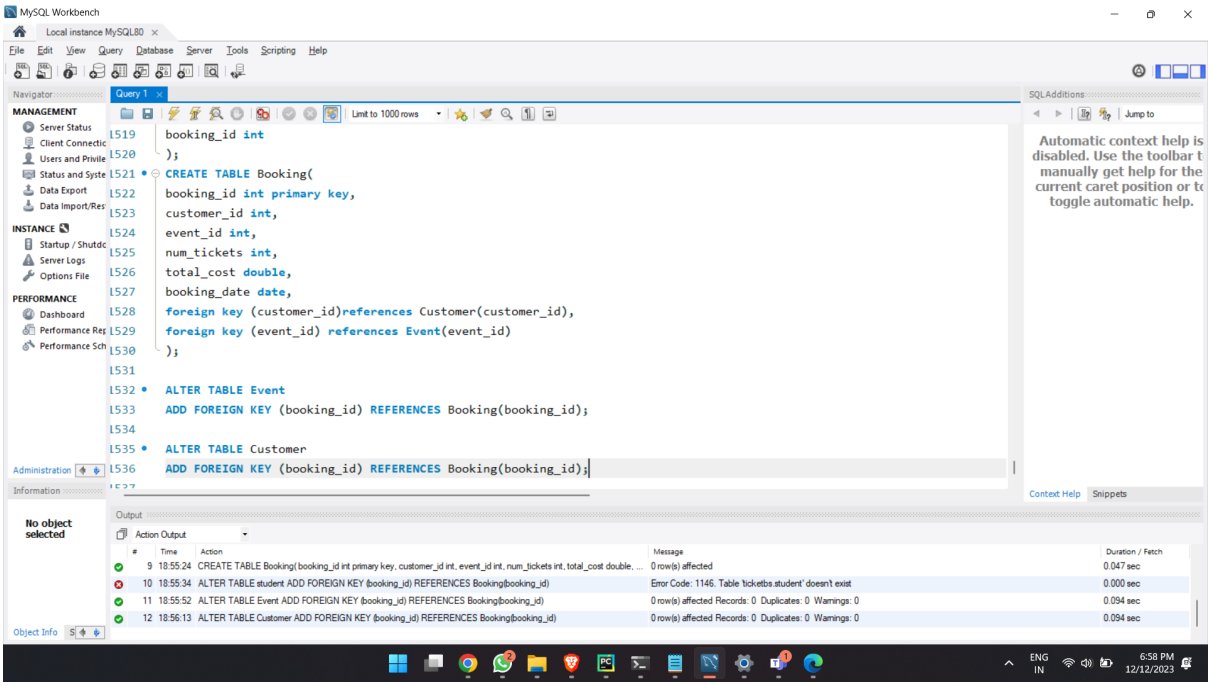
- **Venu**
- **Event**
- **Customers**
- **Booking**



Q3. Create an ERD (Entity Relationship Diagram) for the database.

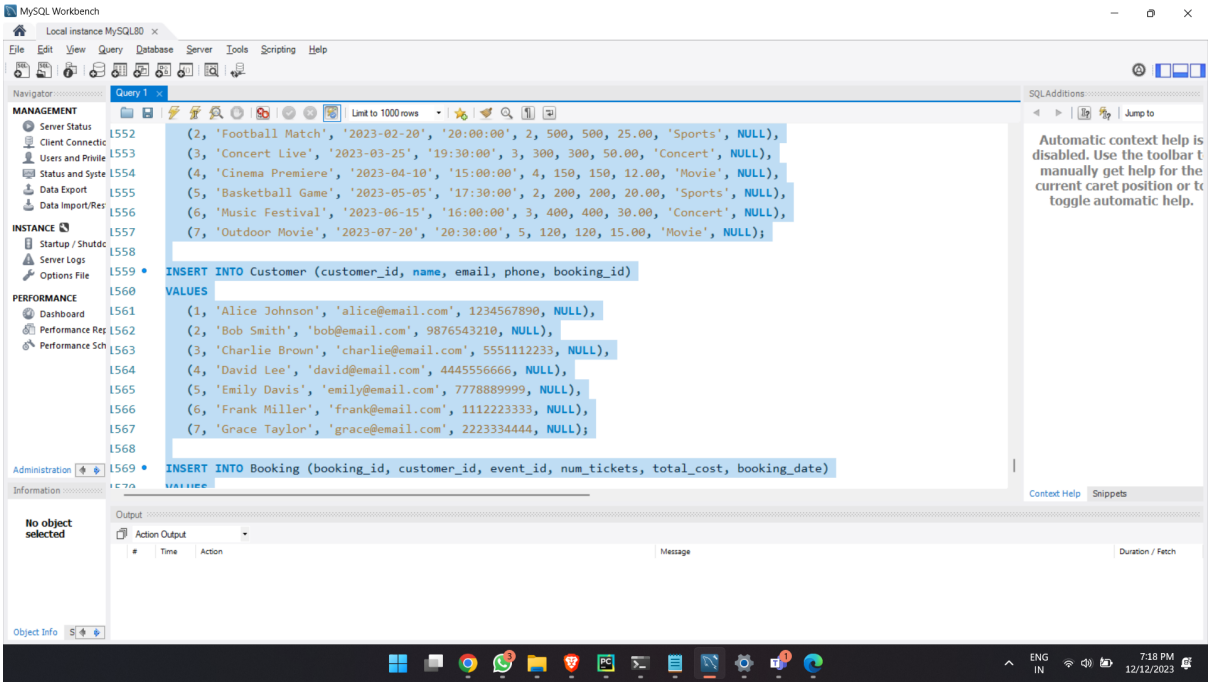


Q4. Create appropriate Primary Key and Foreign Key constraints for referential integrity

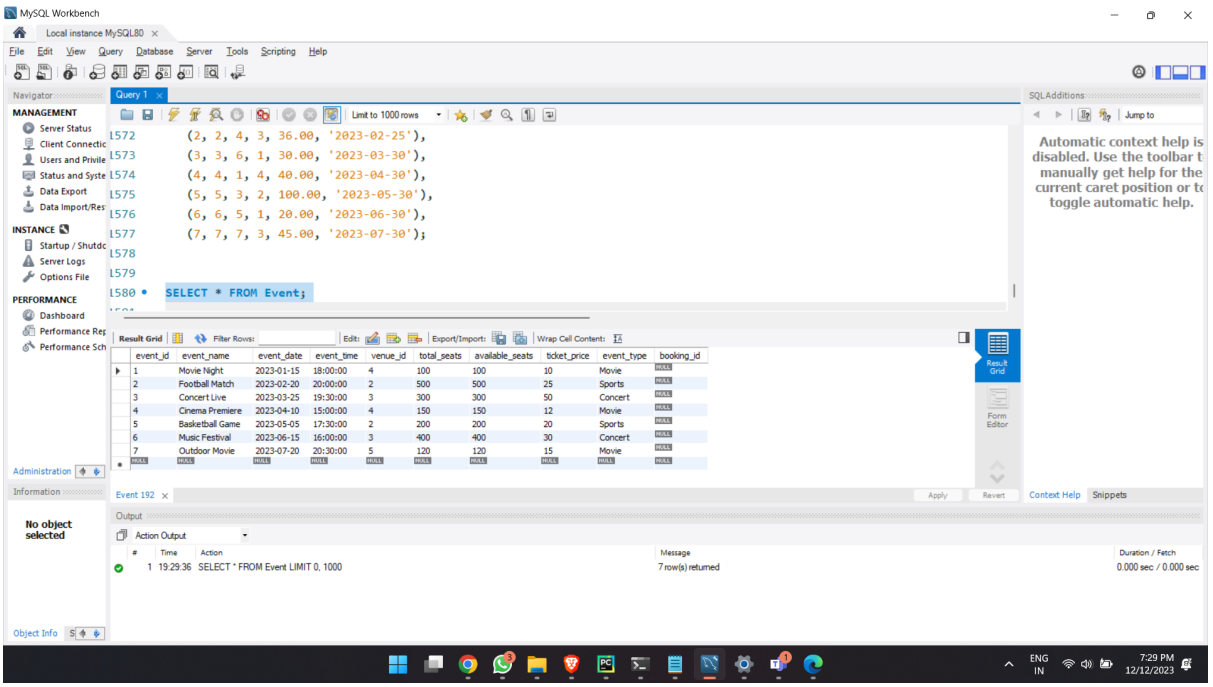


Tasks 2: Select, Where, Between, AND, LIKE:

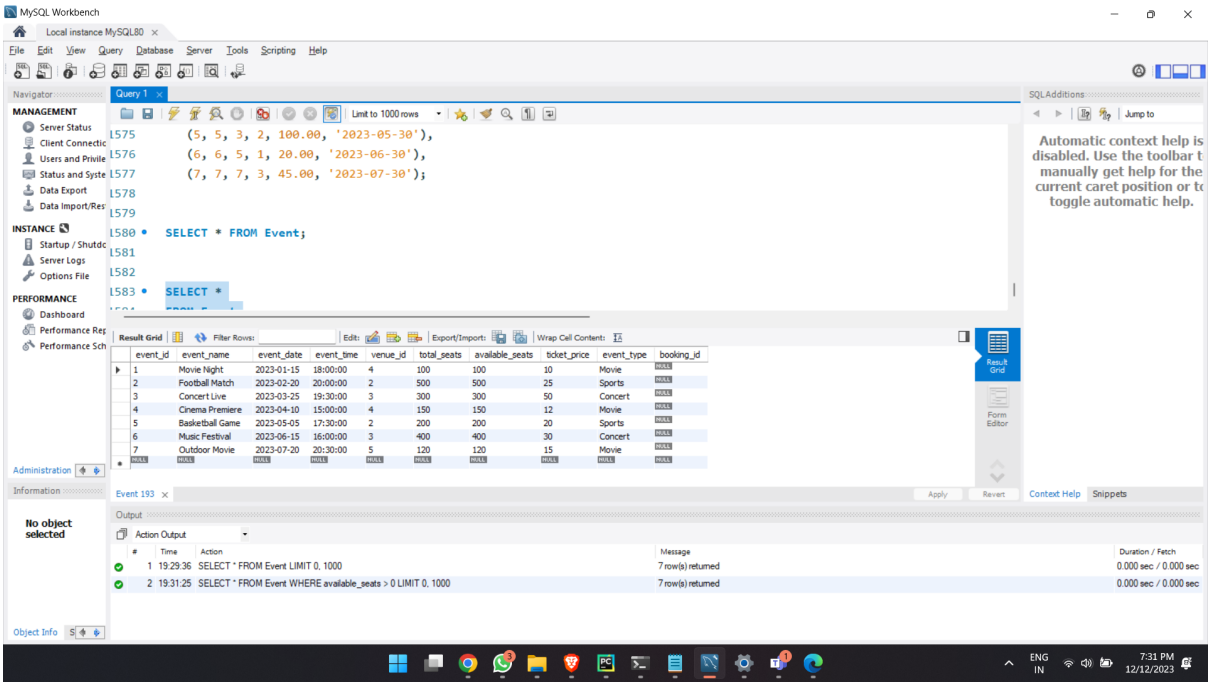
Q1. Write a SQL query to insert at least 10 sample records into each table.



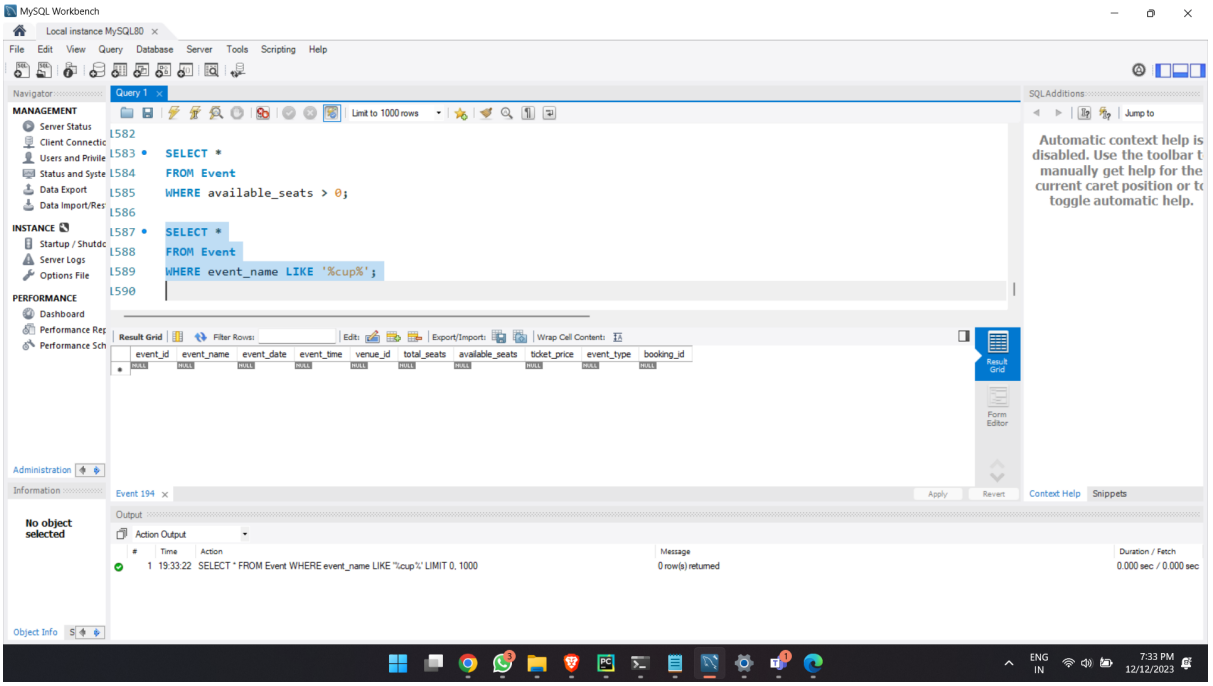
Q2. Write a SQL query to list all Events.



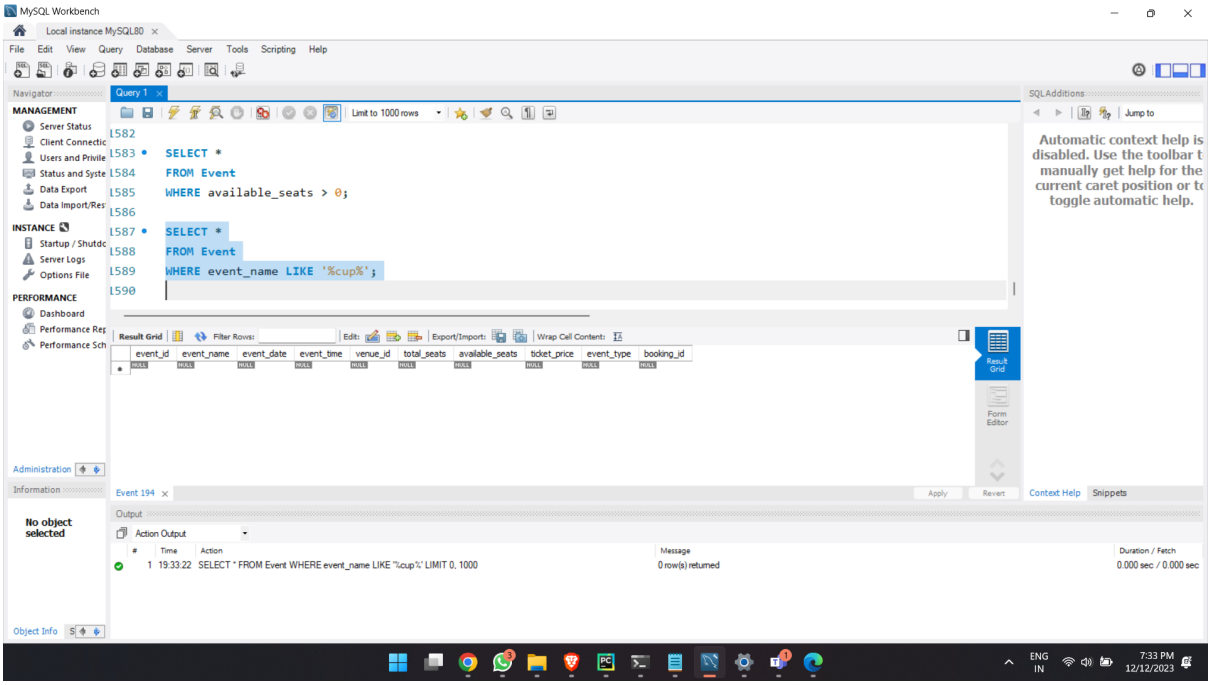
Q3. Write a SQL query to select events with available tickets.



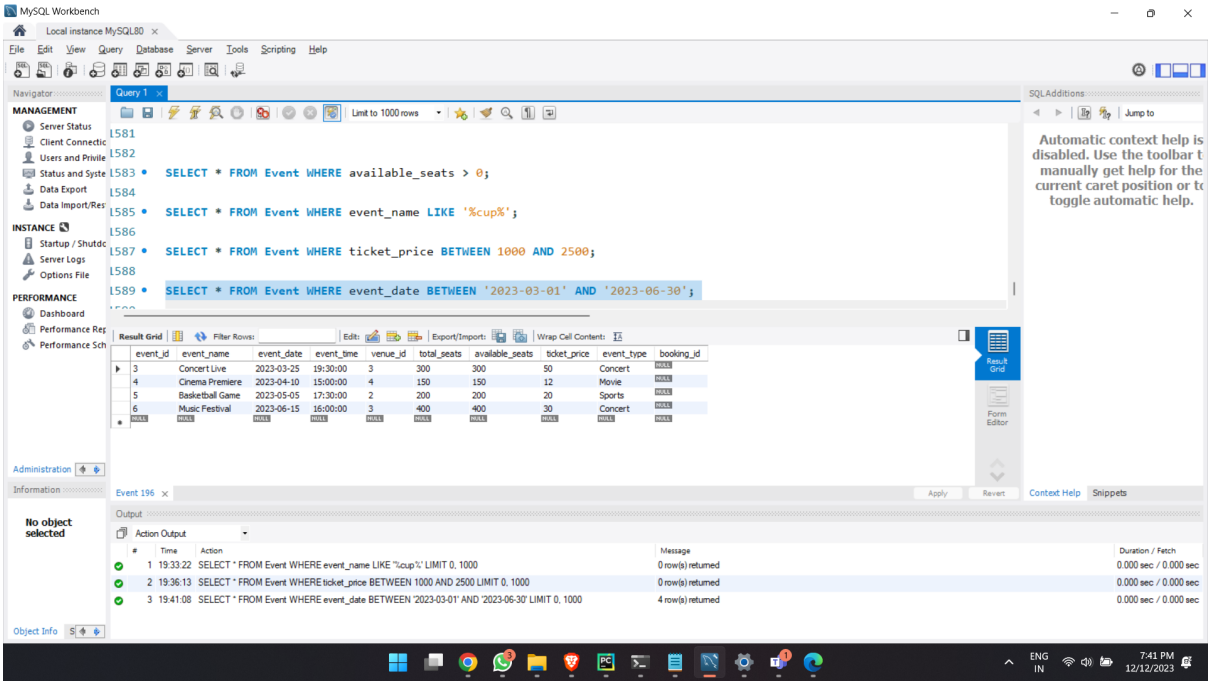
Q4. Write a SQL query to select events name partial match with ‘cup’.



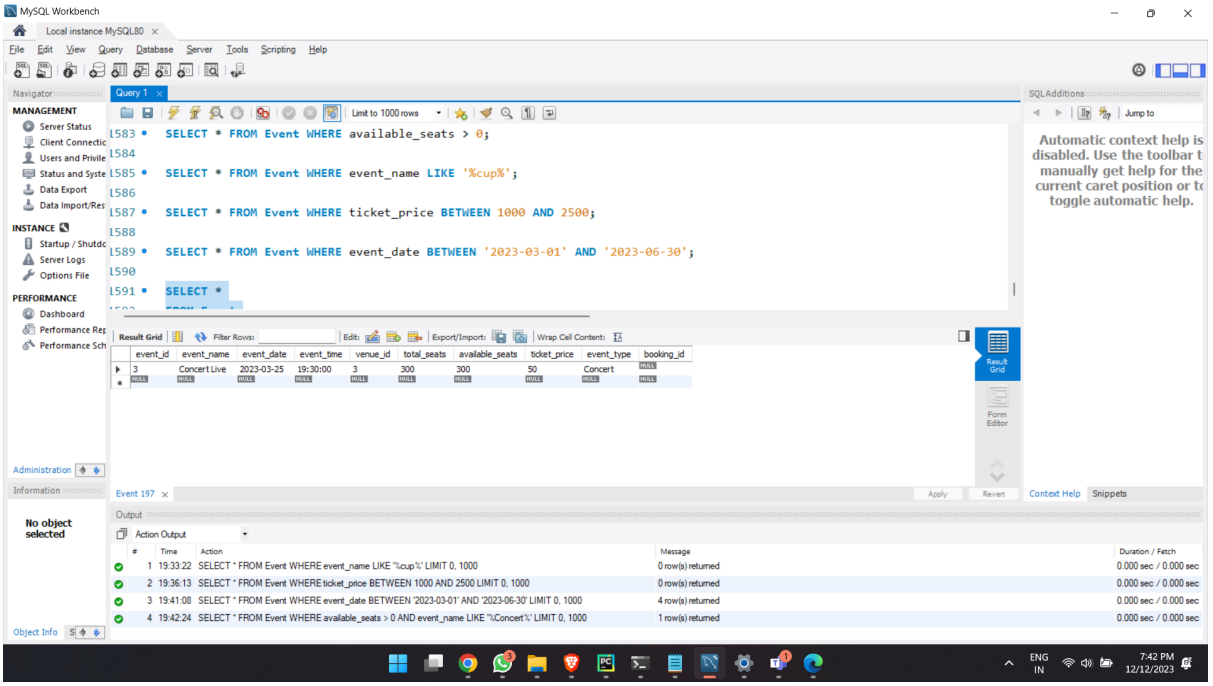
Q5. Write a SQL query to select events with ticket price range is between 1000 to 2500.



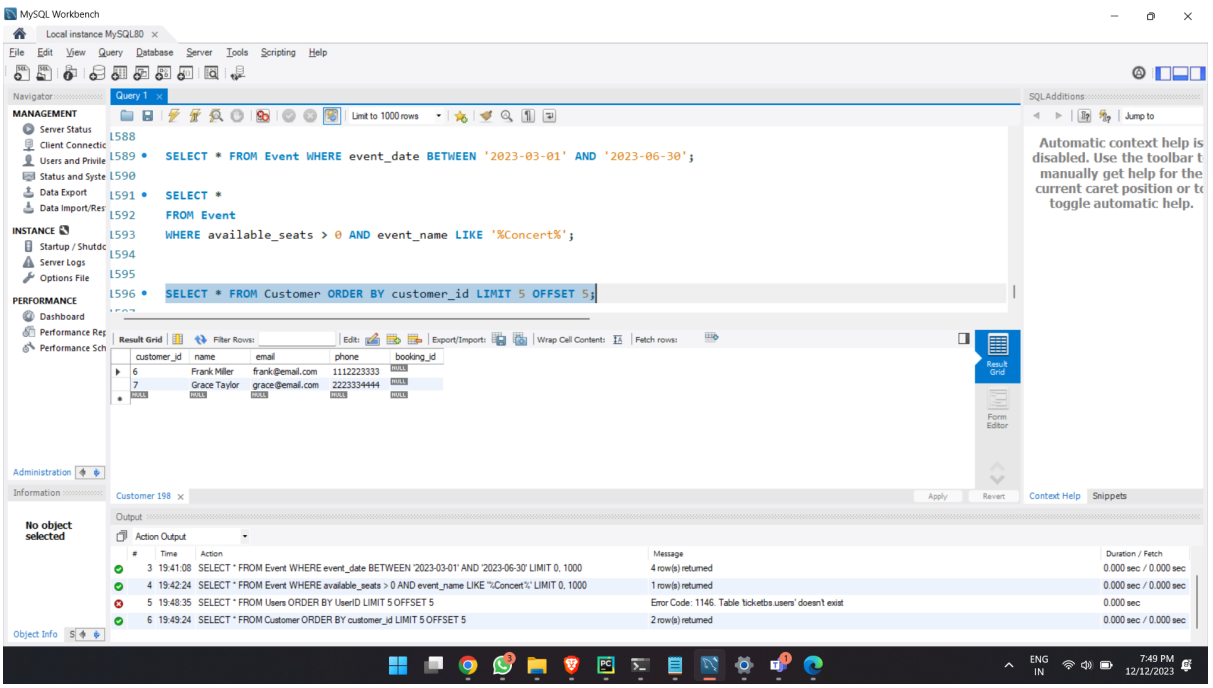
Q6. Write a SQL query to retrieve events with dates falling within a specific range.



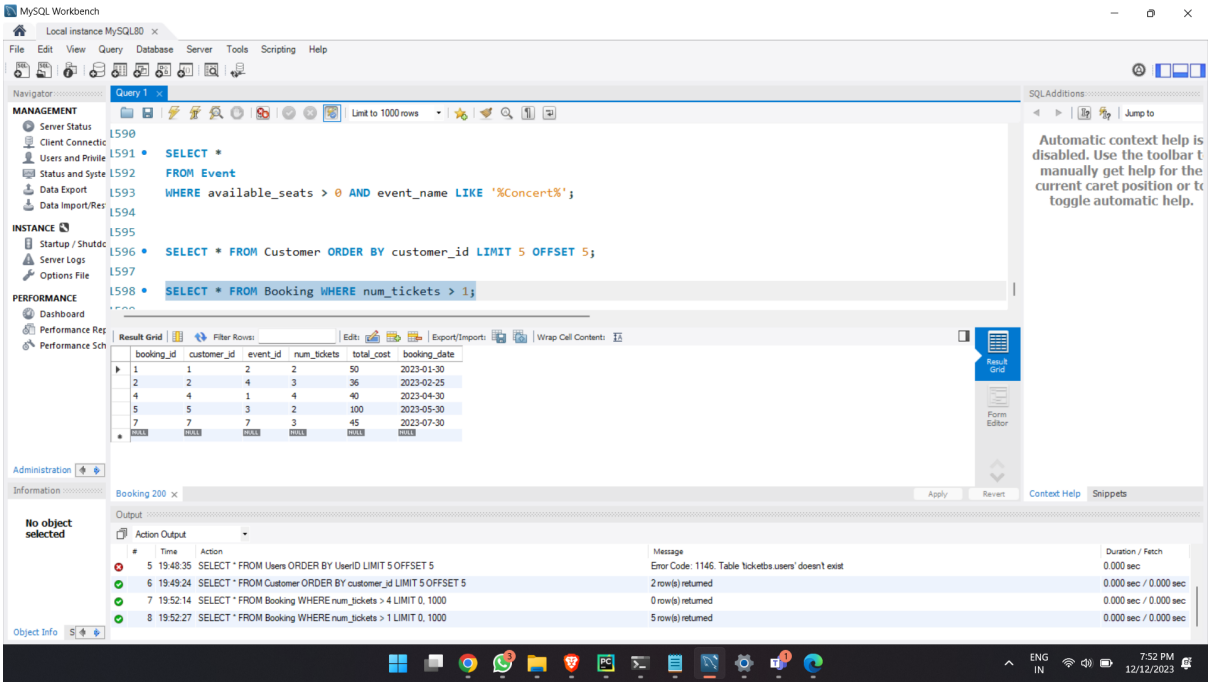
Q7. Write a SQL query to retrieve events with available tickets that also have "Concert" in their name.



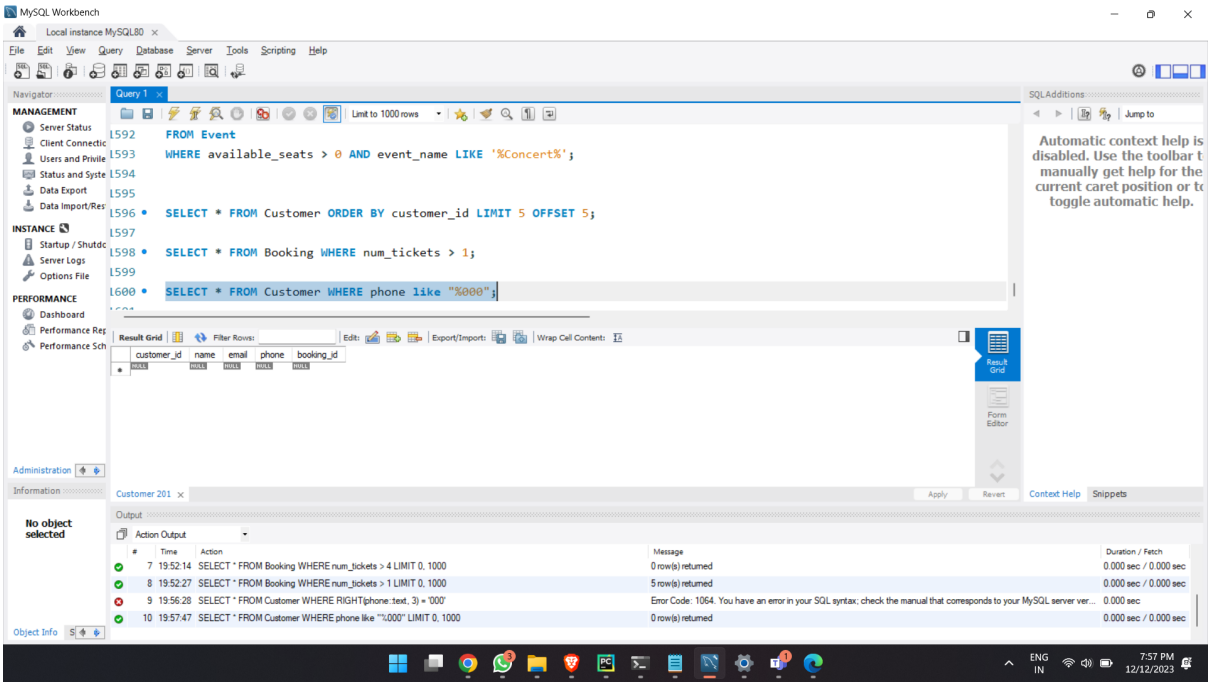
Q8. Write a SQL query to retrieve users in batches of 5, starting from the 6th user.



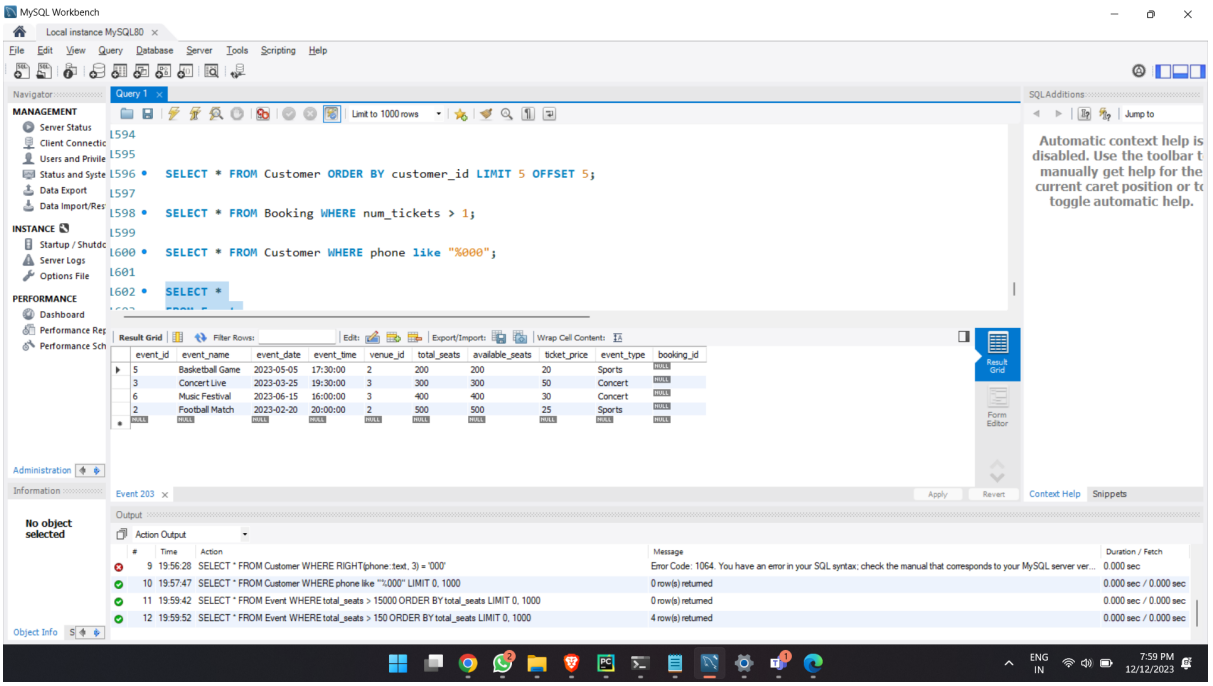
Q9. Write a SQL query to retrieve bookings details contains booked no of ticket more than 1.



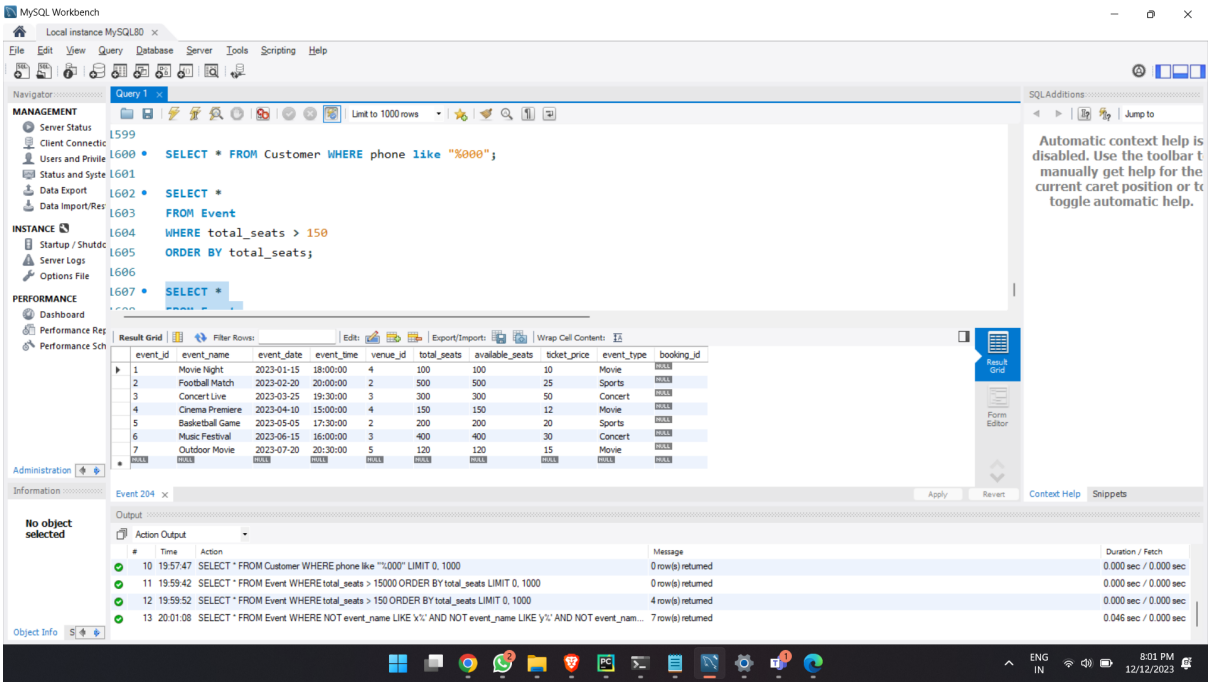
Q10. Write a SQL query to retrieve customer information whose phone number end with ‘000’



Q11. Write a SQL query to retrieve the events in order whose seat capacity more than 150.

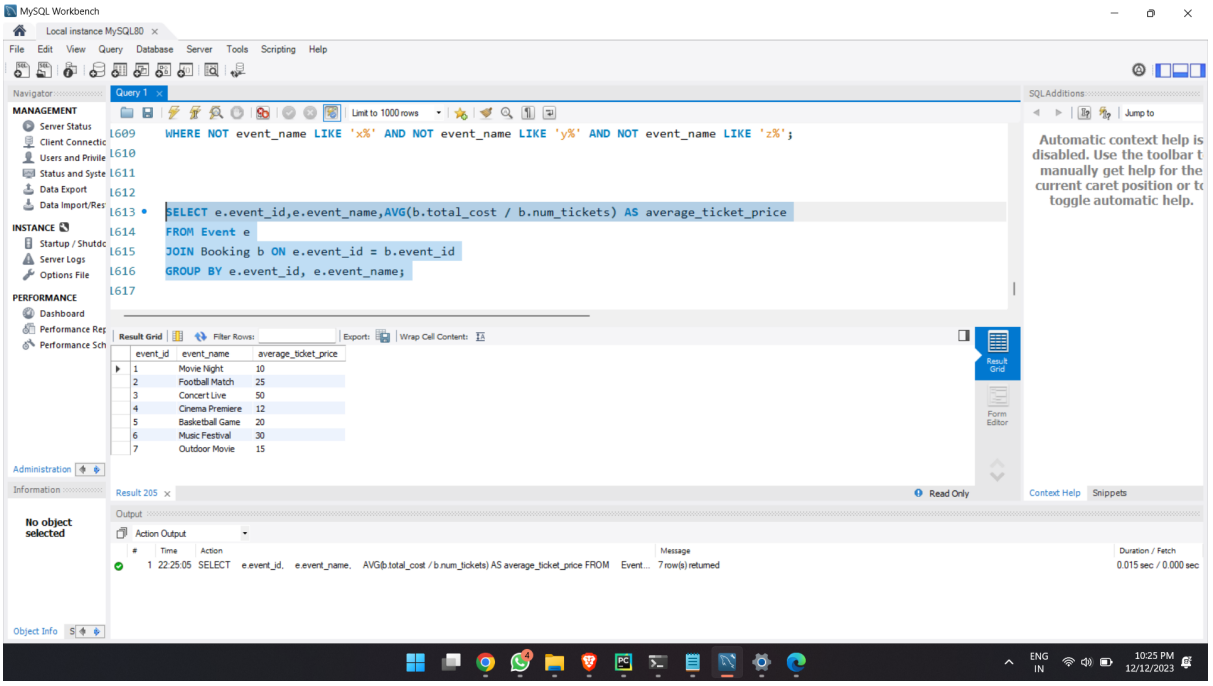


Q12. Write a SQL query to select events name not start with ‘x’, ‘y’, ‘z’

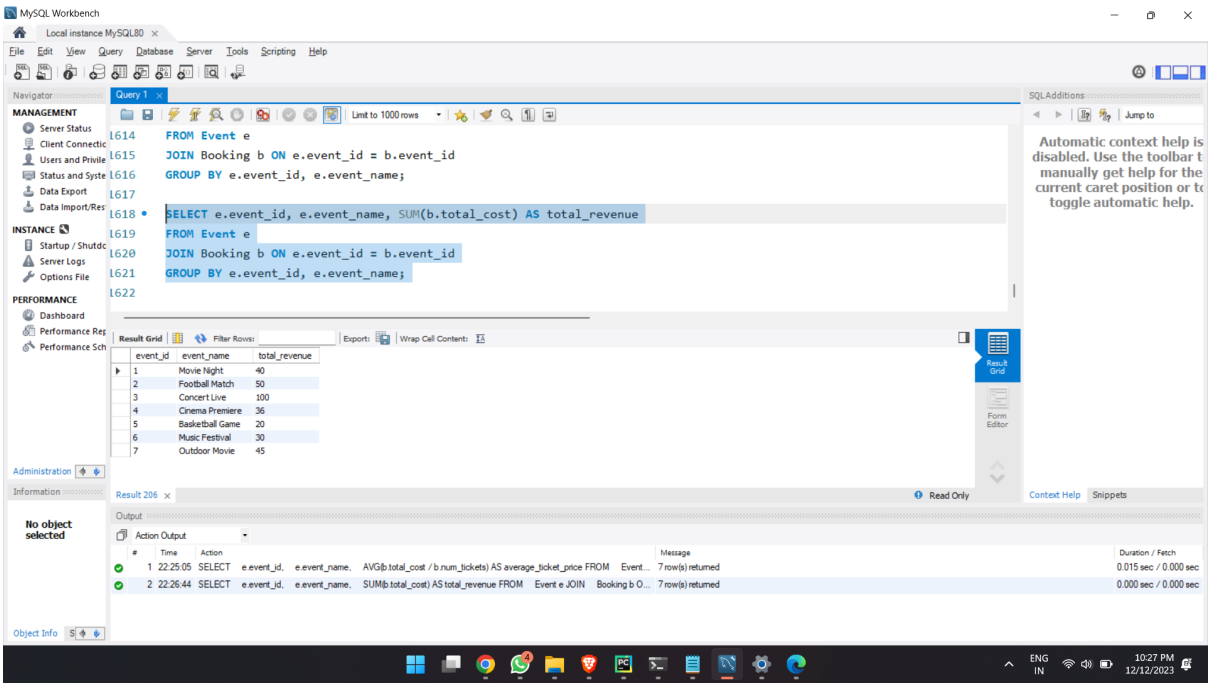


Tasks 3: Aggregate functions, Having, Order By, GroupBy and Joins:

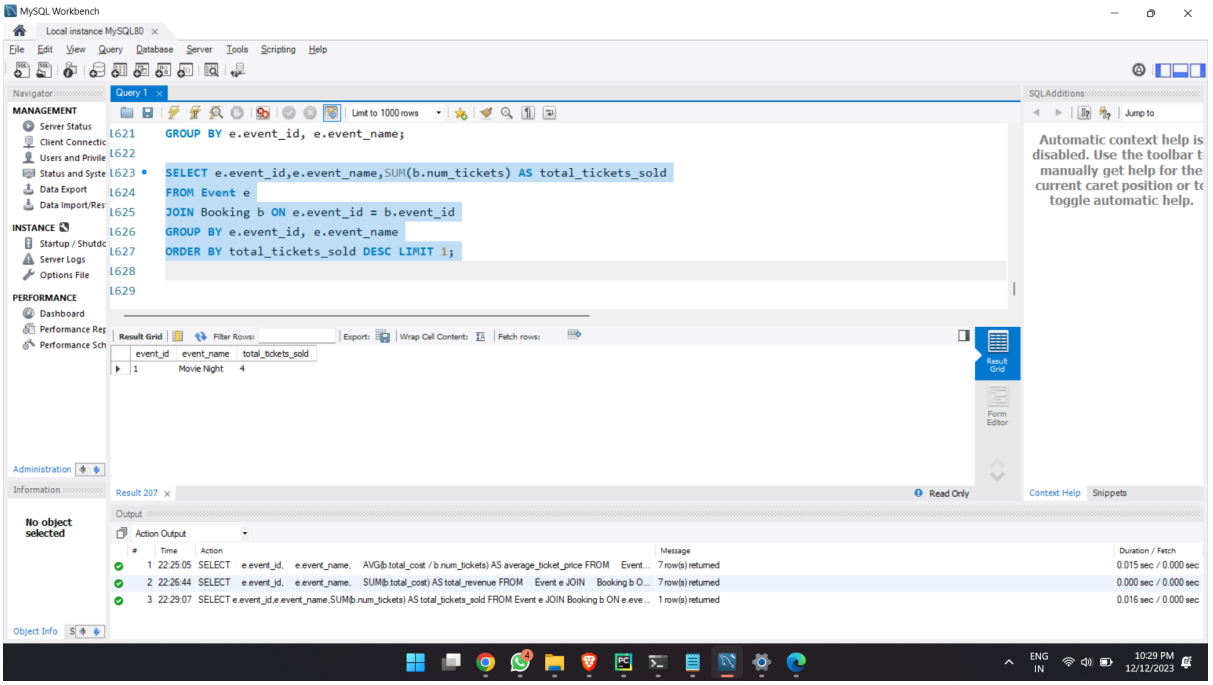
Q1. Write a SQL query to List Events and Their Average Ticket Prices.



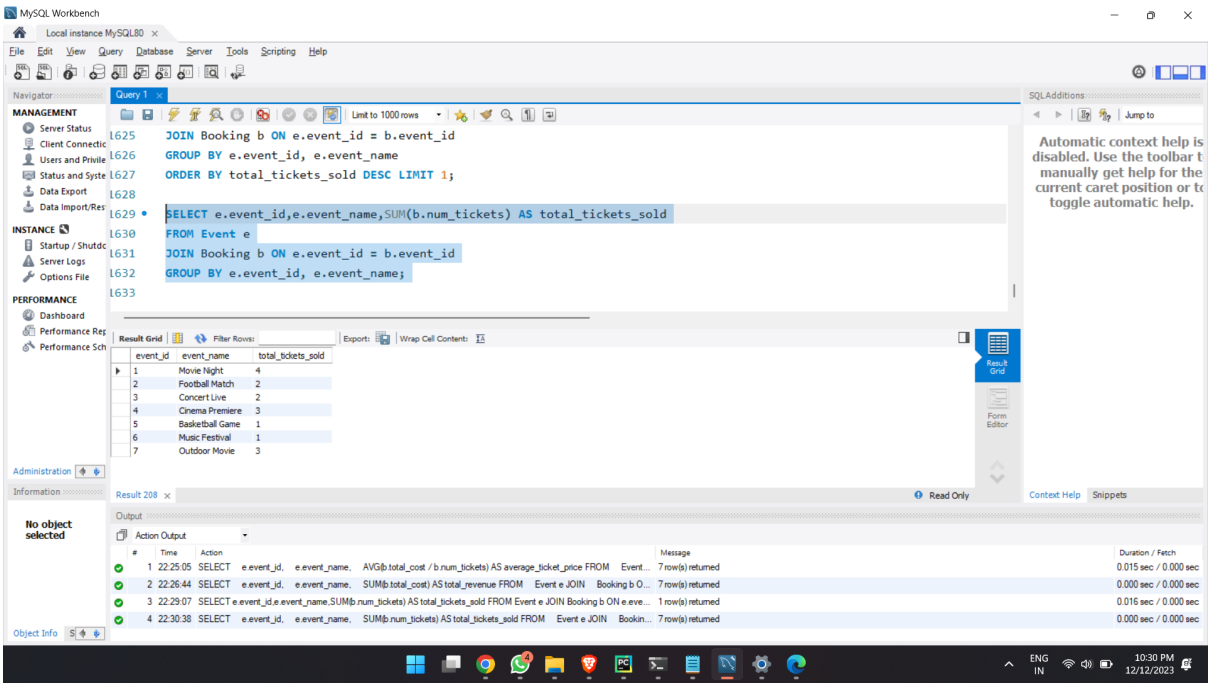
Q2. Write a SQL query to Calculate the Total Revenue Generated by Events.



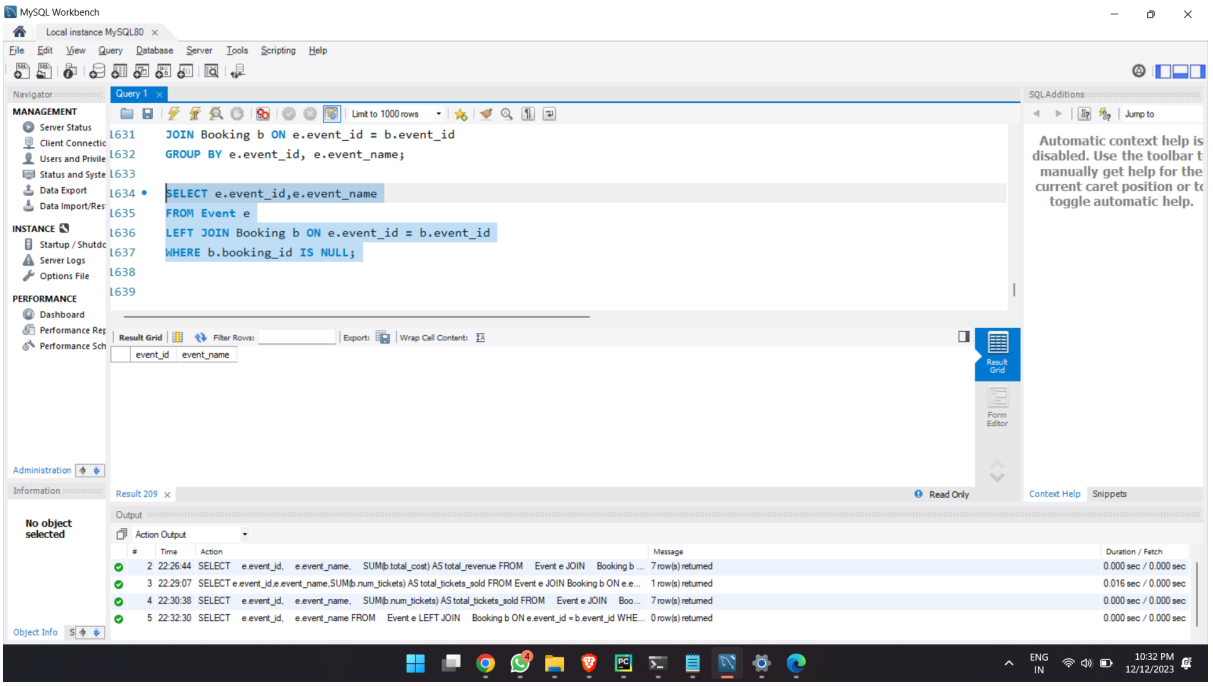
Q3. Write a SQL query to find the event with the highest ticket sales.



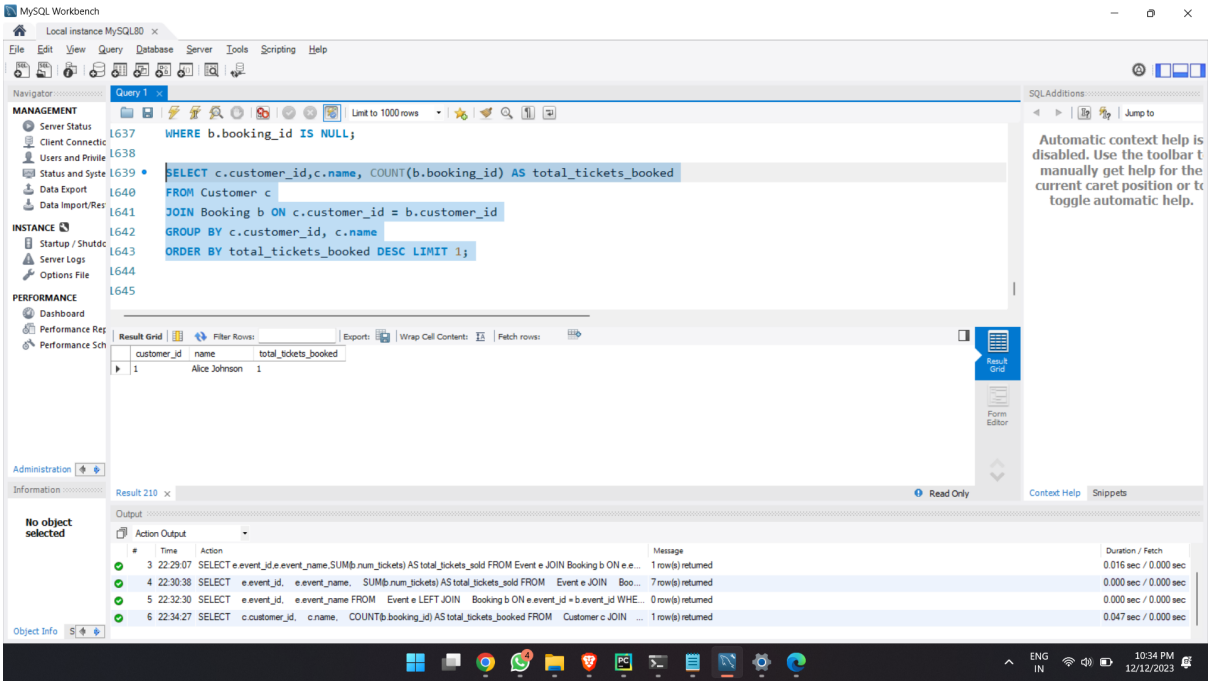
Q4. Write a SQL query to Calculate the Total Number of Tickets Sold for Each Event.



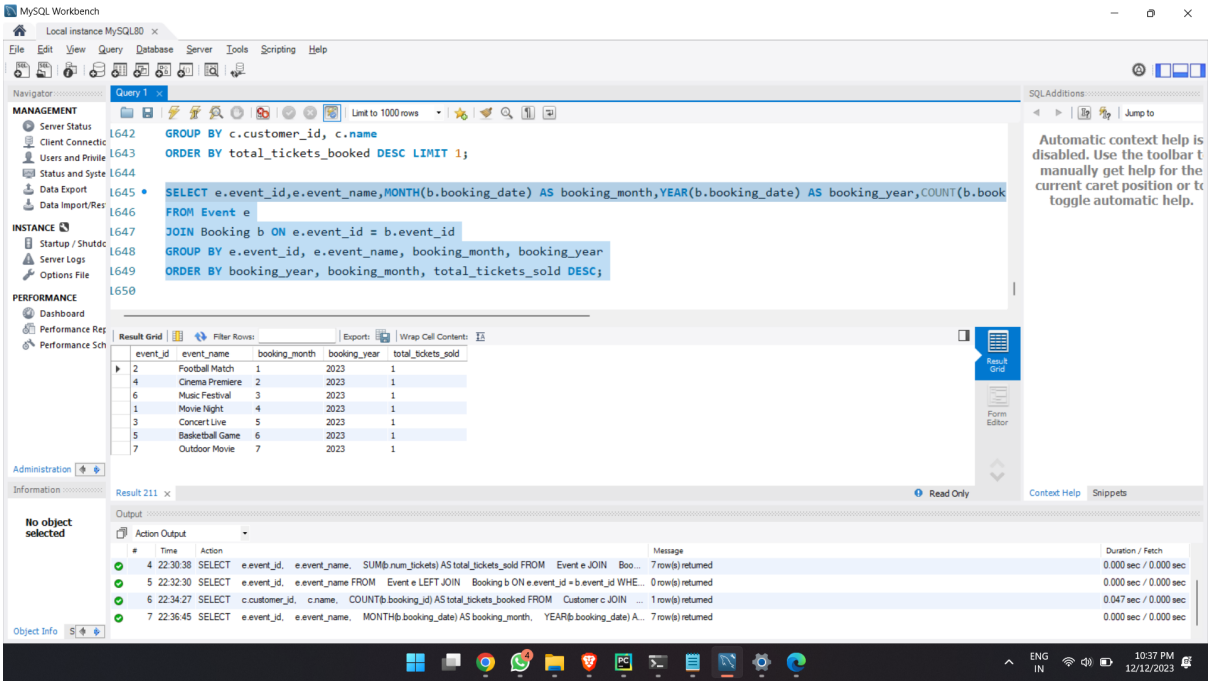
Q5. Write a SQL query to Find Events with No Ticket Sales.



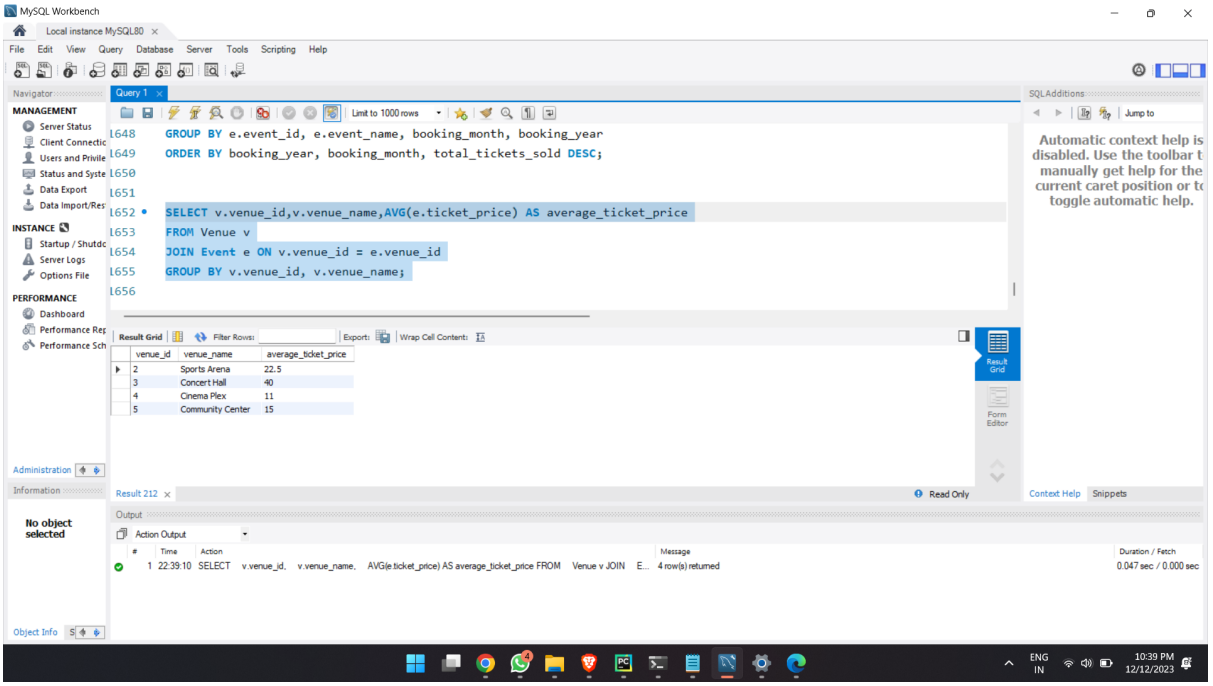
Q6. Write a SQL query to Find the User Who Has Booked the Most Tickets.



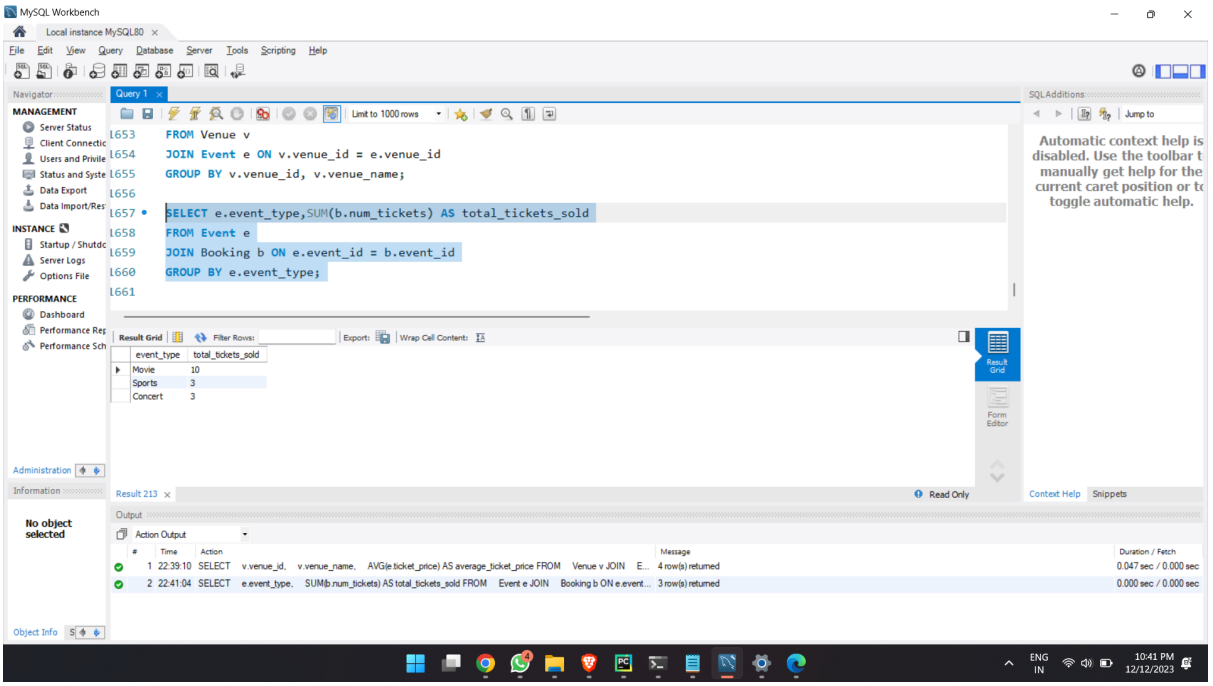
Q7. Write a SQL query to List Events and the total number of tickets sold for each month.



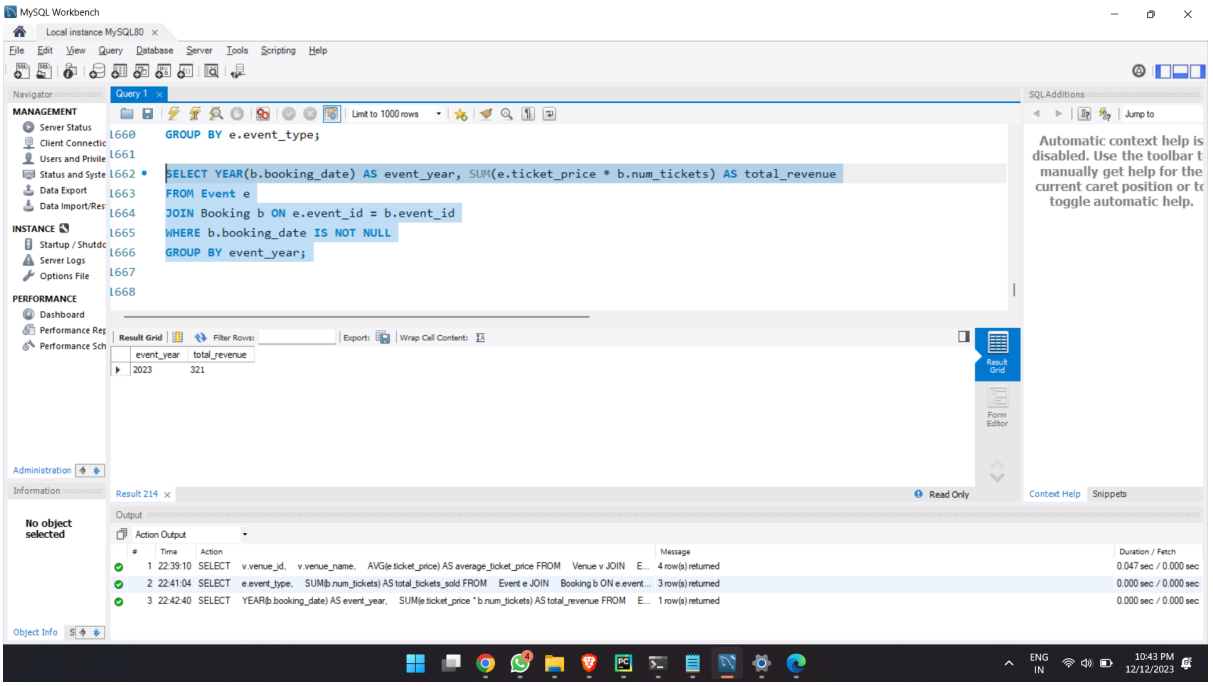
Q8. Write a SQL query to calculate the average Ticket Price for Events in Each Venue.



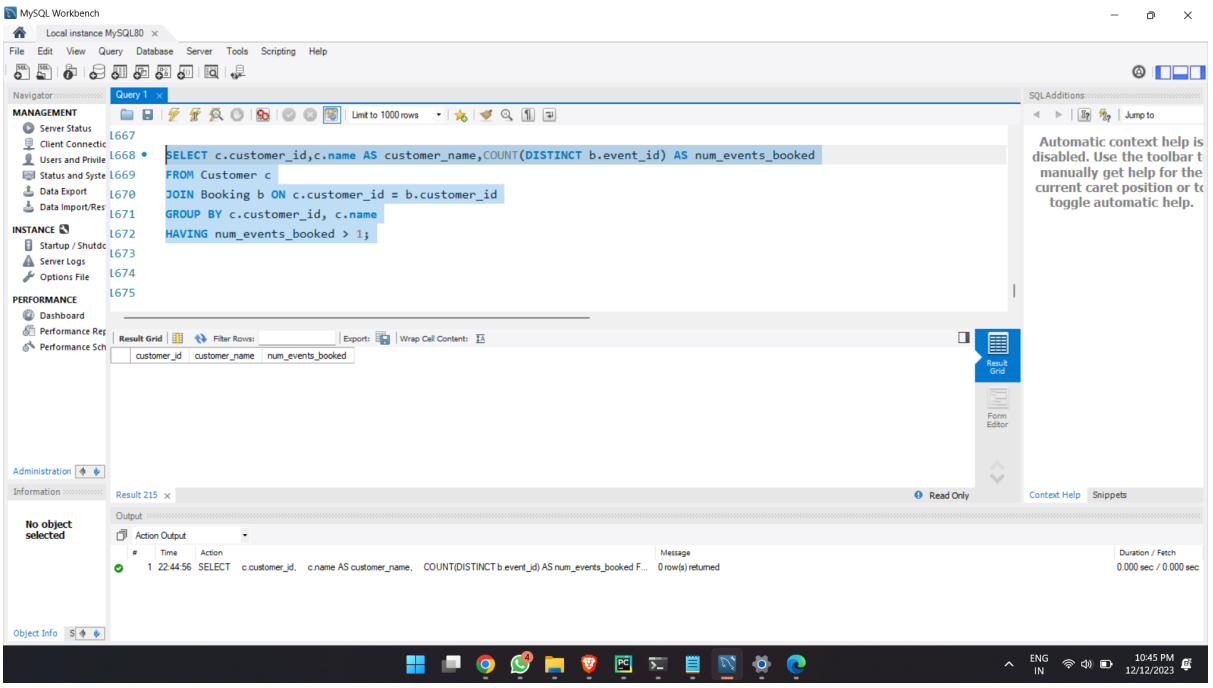
Q9. Write a SQL query to calculate the total Number of Tickets Sold for Each Event Type.



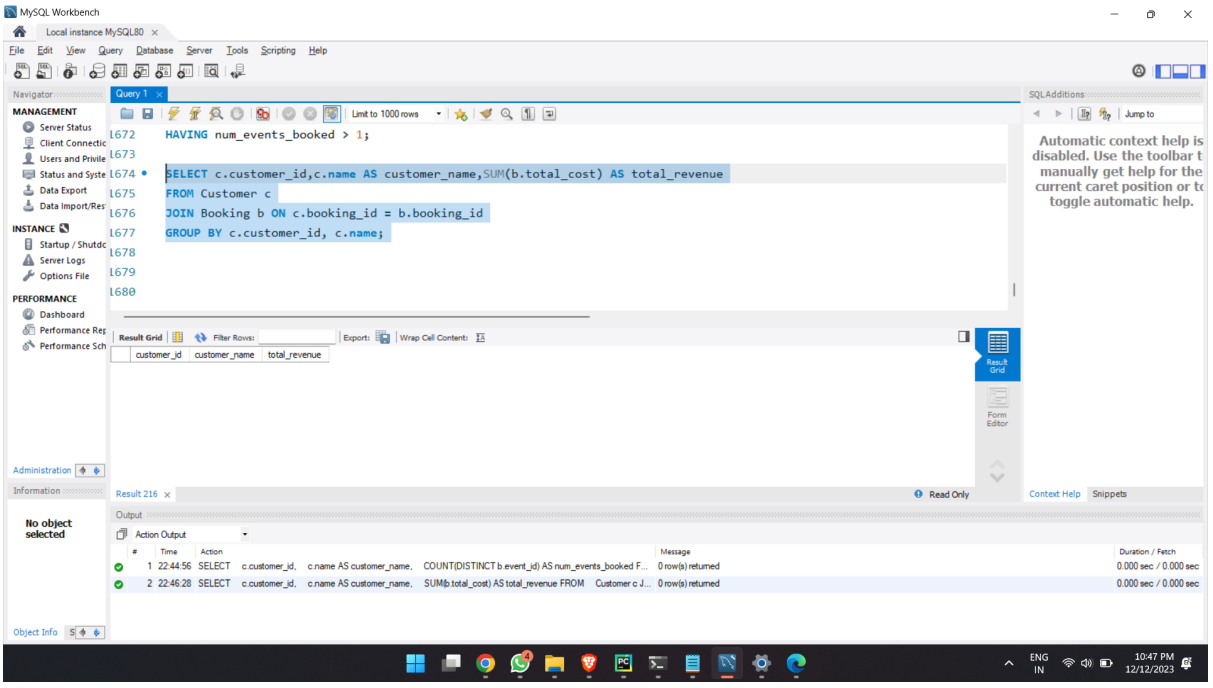
Q10. Write a SQL query to calculate the total Revenue Generated by Events in Each Year.



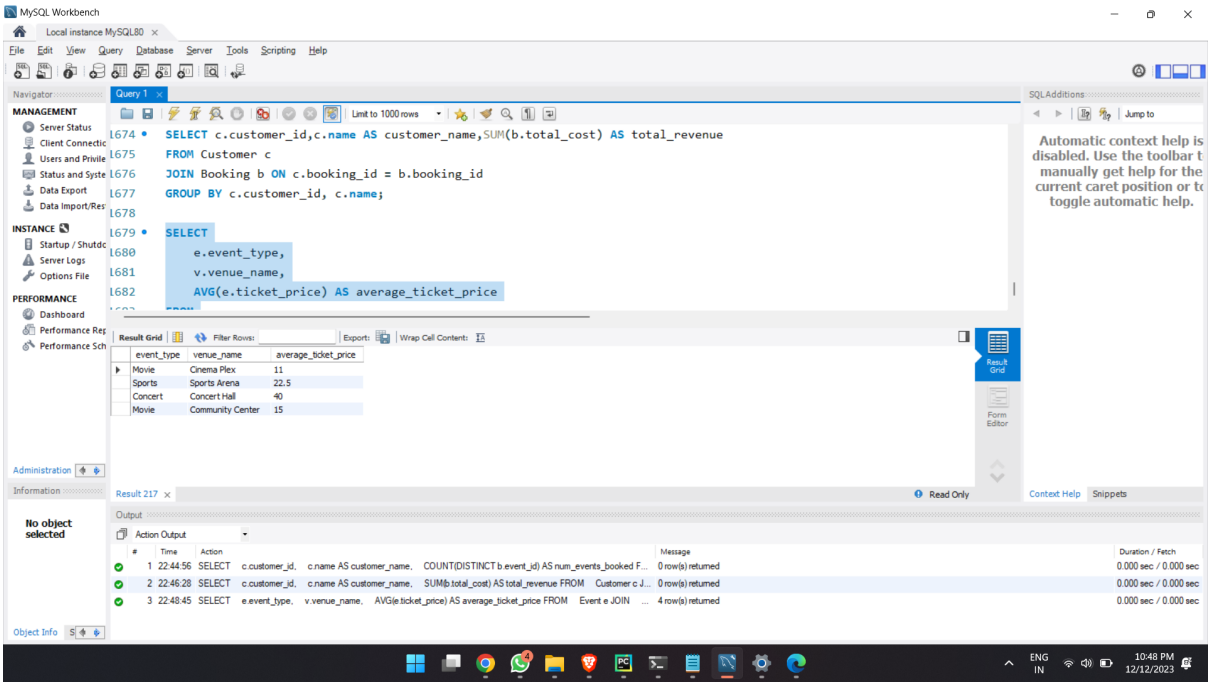
Q11. Write a SQL query to list users who have booked tickets for multiple events.



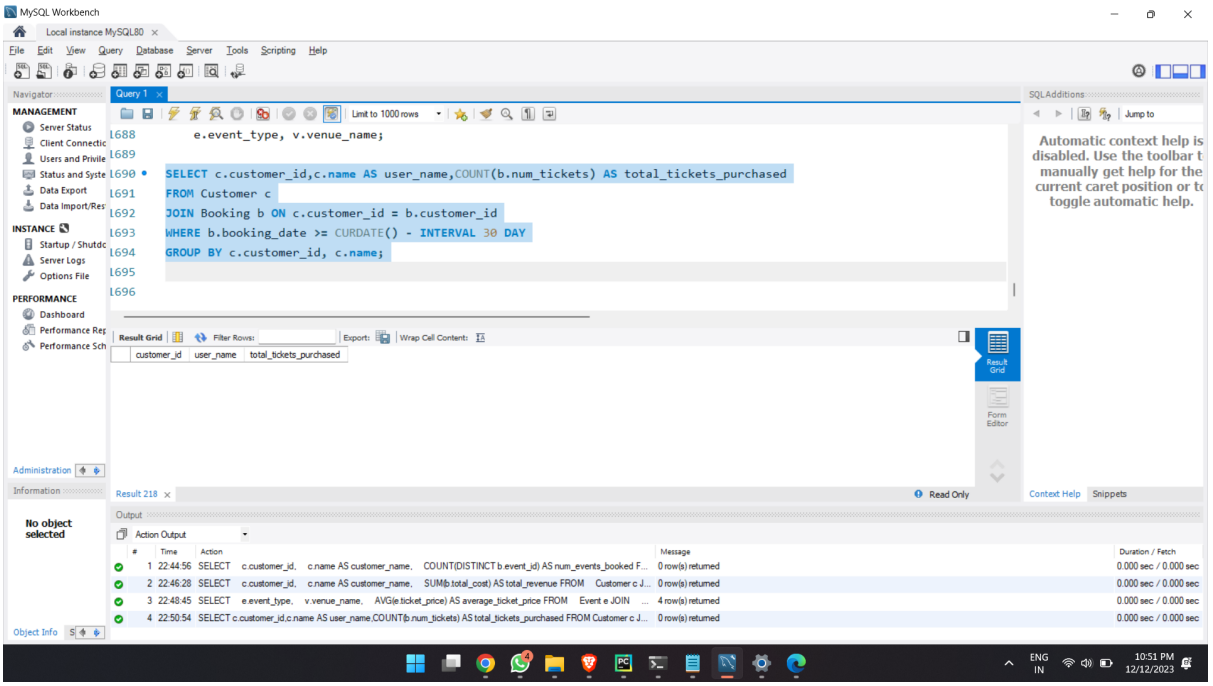
Q12. Write a SQL query to calculate the Total Revenue Generated by Events for Each User.



Q13. Write a SQL query to calculate the Average Ticket Price for Events in Each Category and Venue.

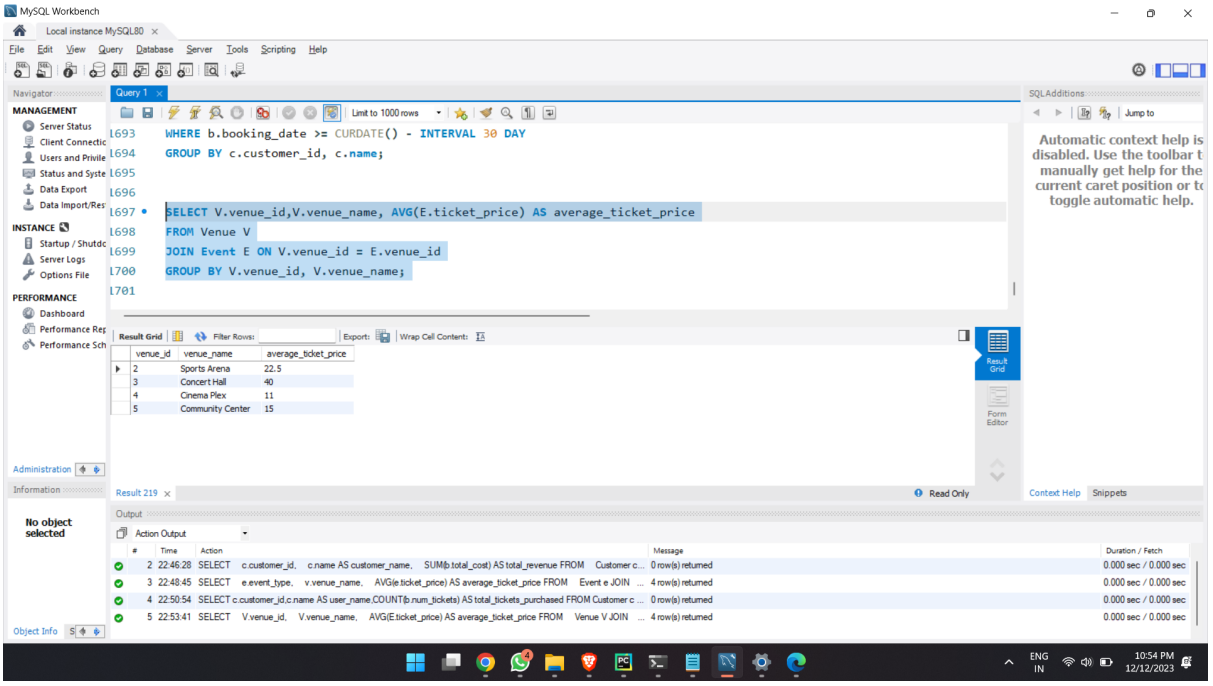


Q14. Write a SQL query to list Users and the Total Number of Tickets They've Purchased in the Last 30 Days.

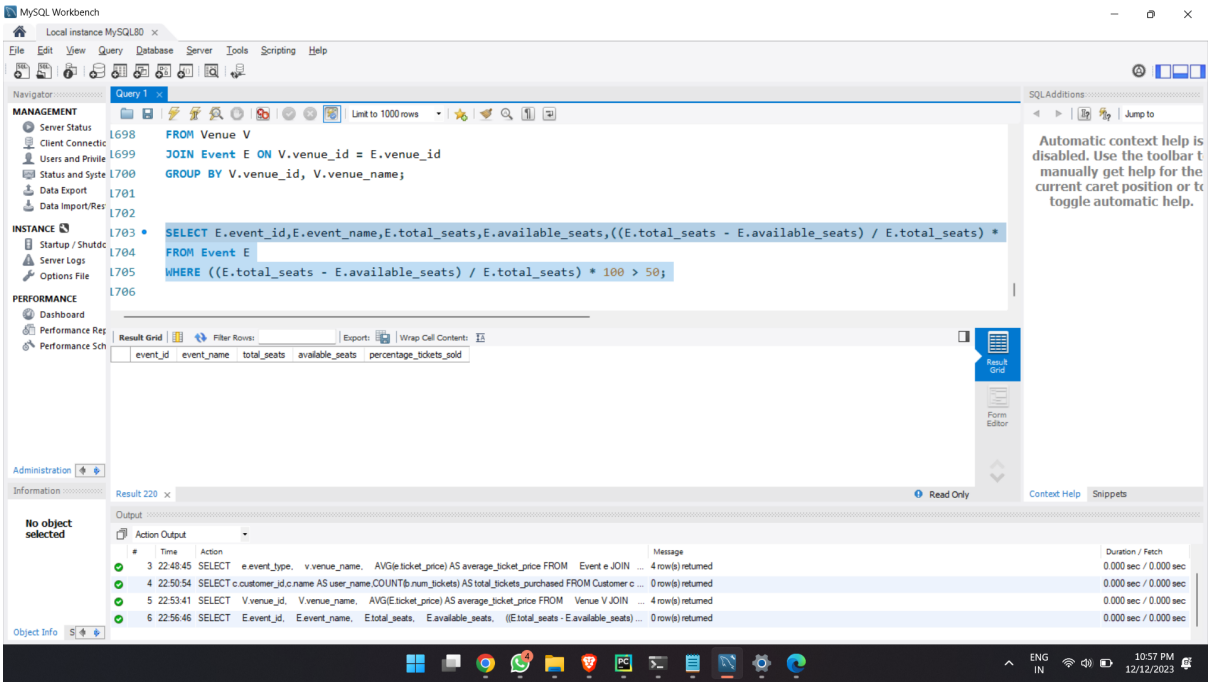


Tasks 4: Subquery and its types

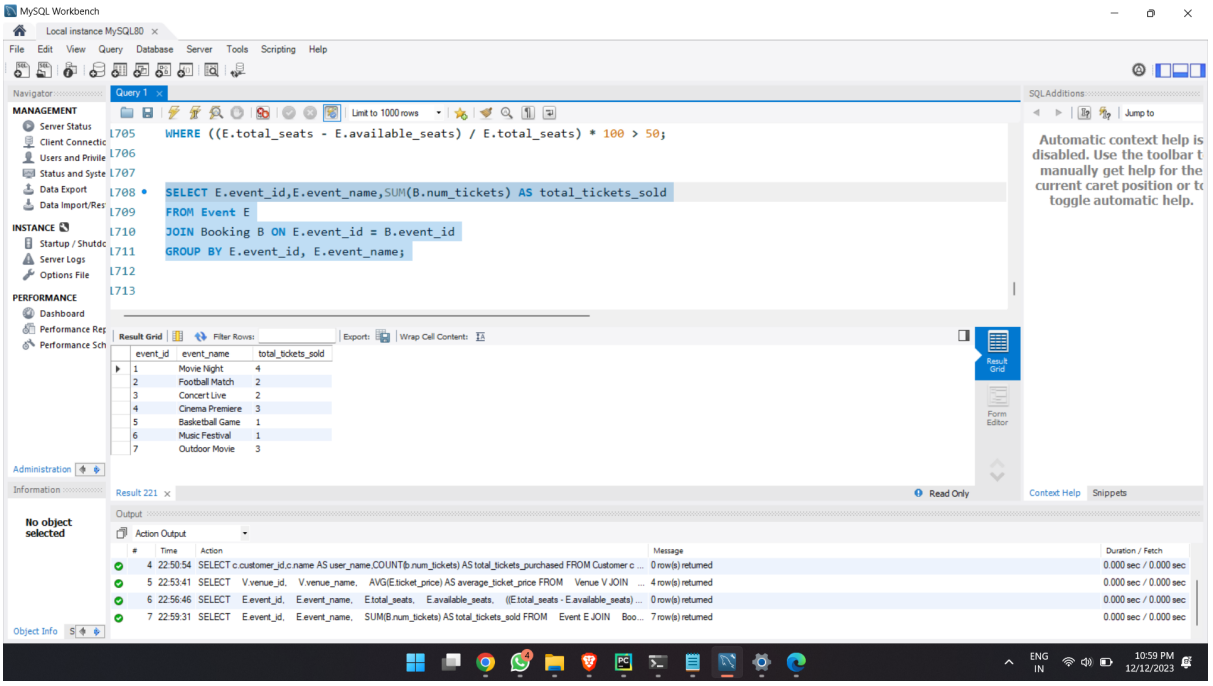
Q1. Calculate the Average Ticket Price for Events in Each Venue Using a Subquery.



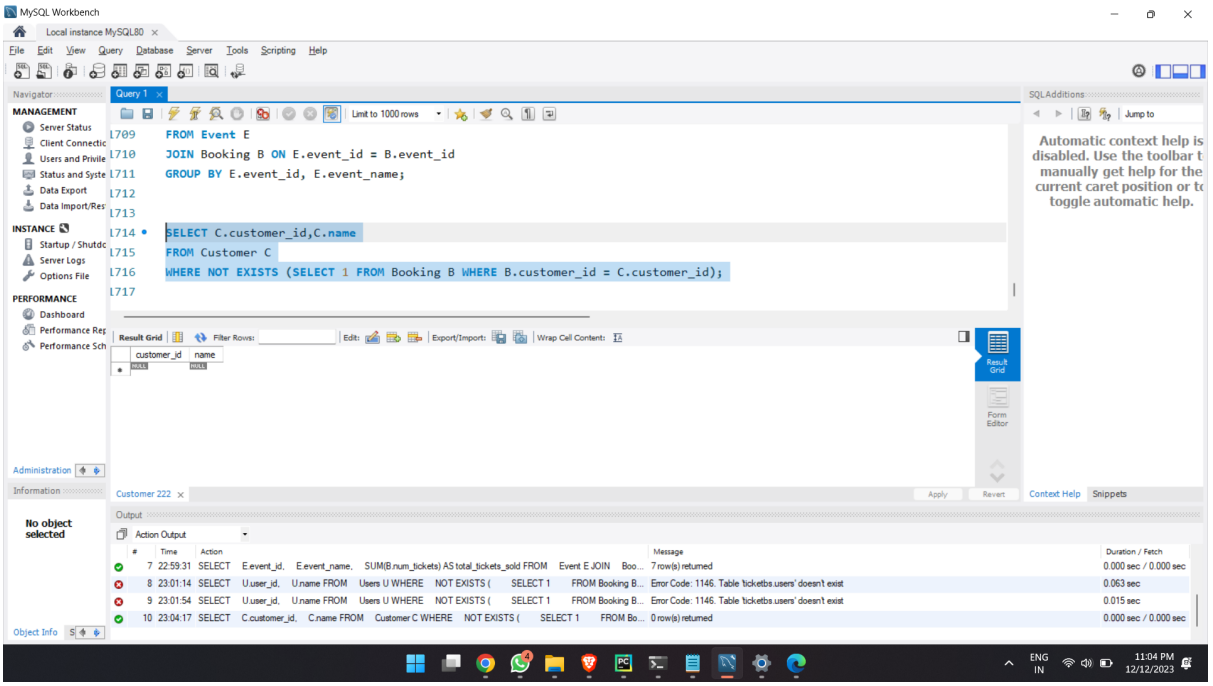
Q2. Find Events with More Than 50% of Tickets Sold using subquery.



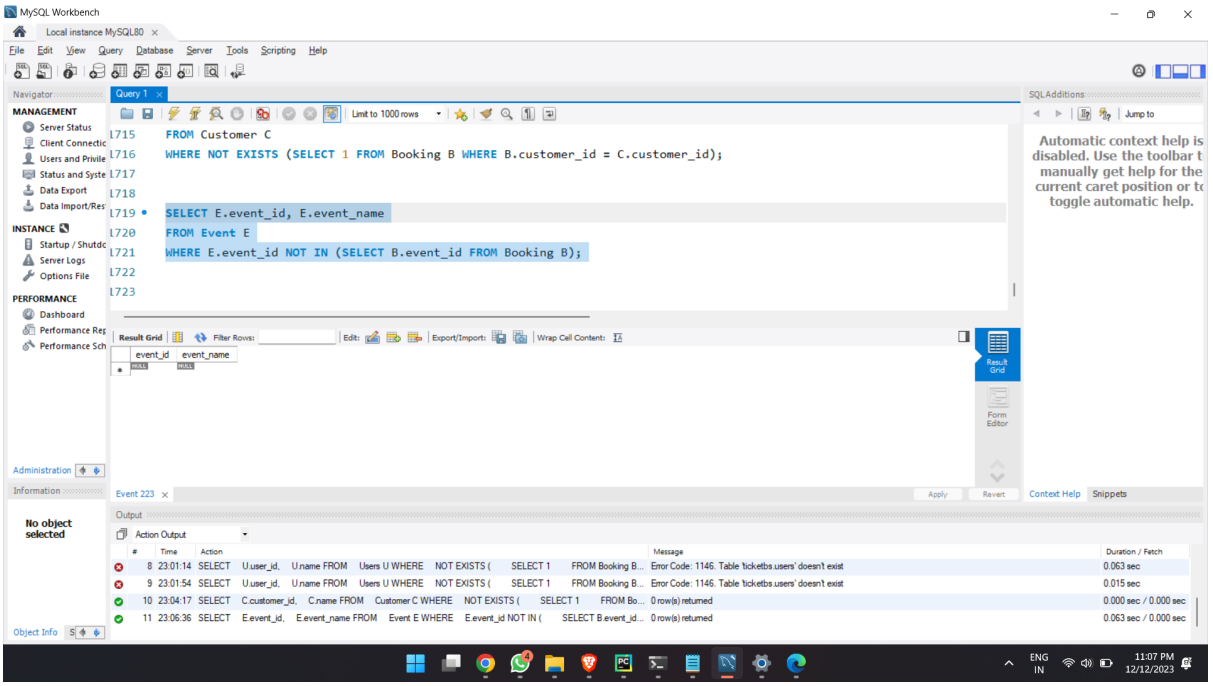
Q3. Calculate the Total Number of Tickets Sold for Each Event.



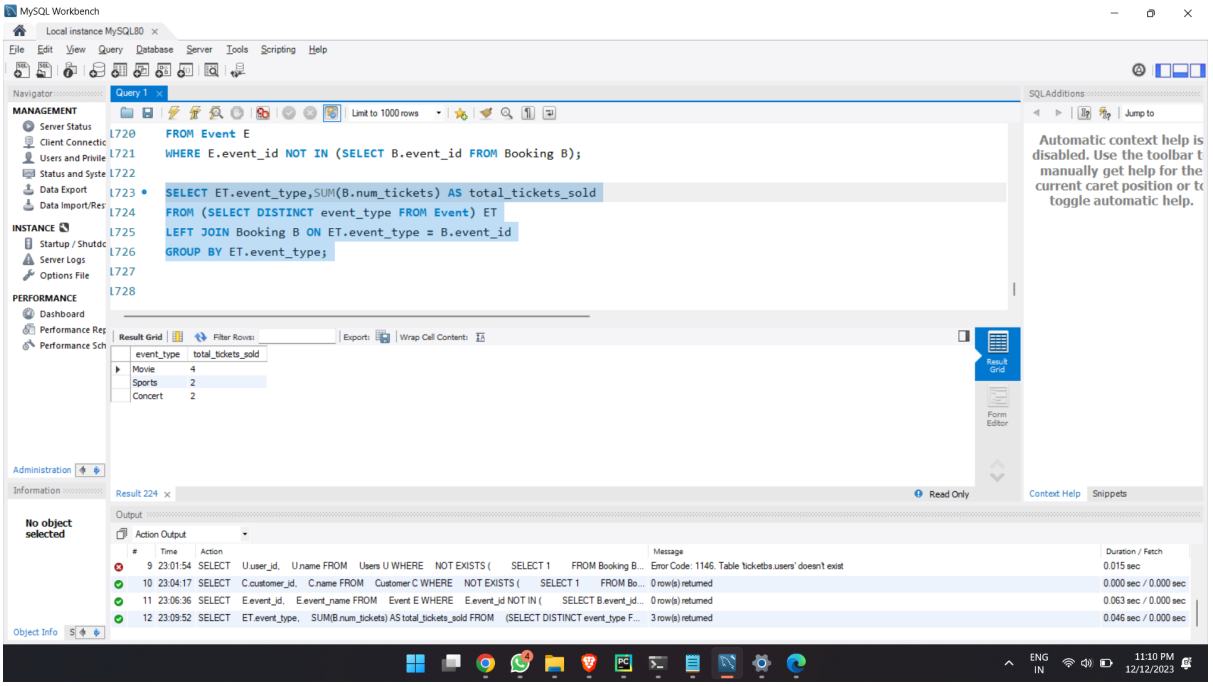
Q4. Find Users Who Have Not Booked Any Tickets Using a NOT EXISTS Subquery.



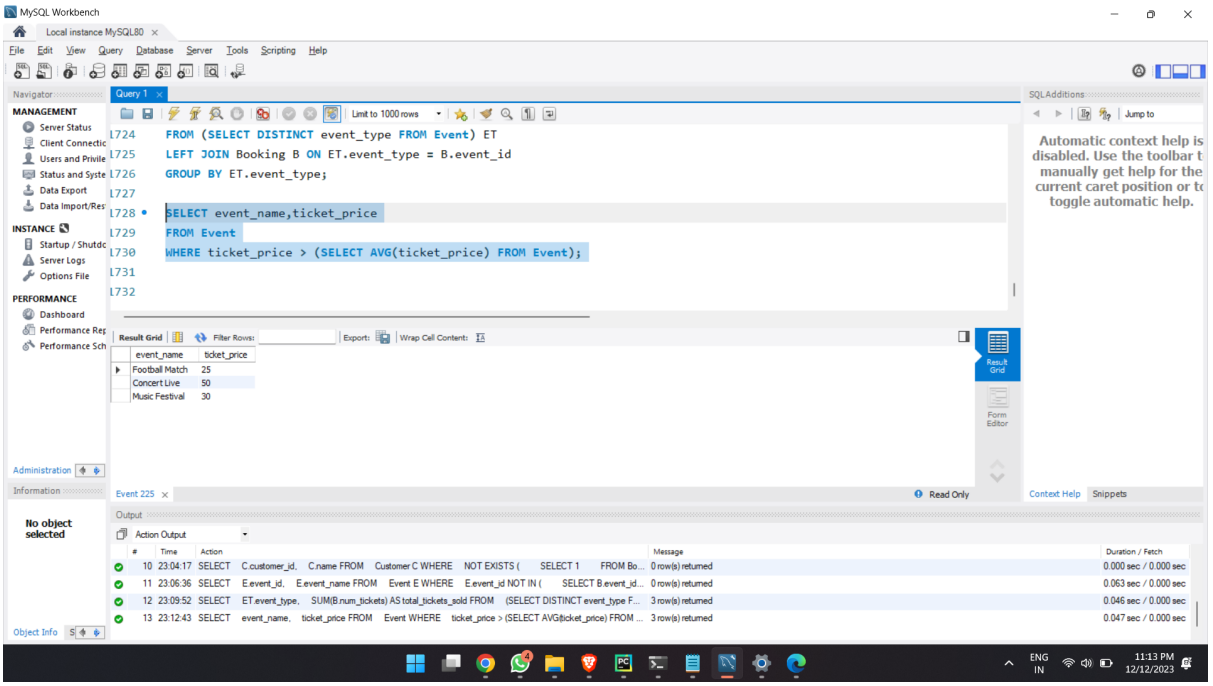
Q5. List Events with No Ticket Sales Using a NOT IN Subquery.



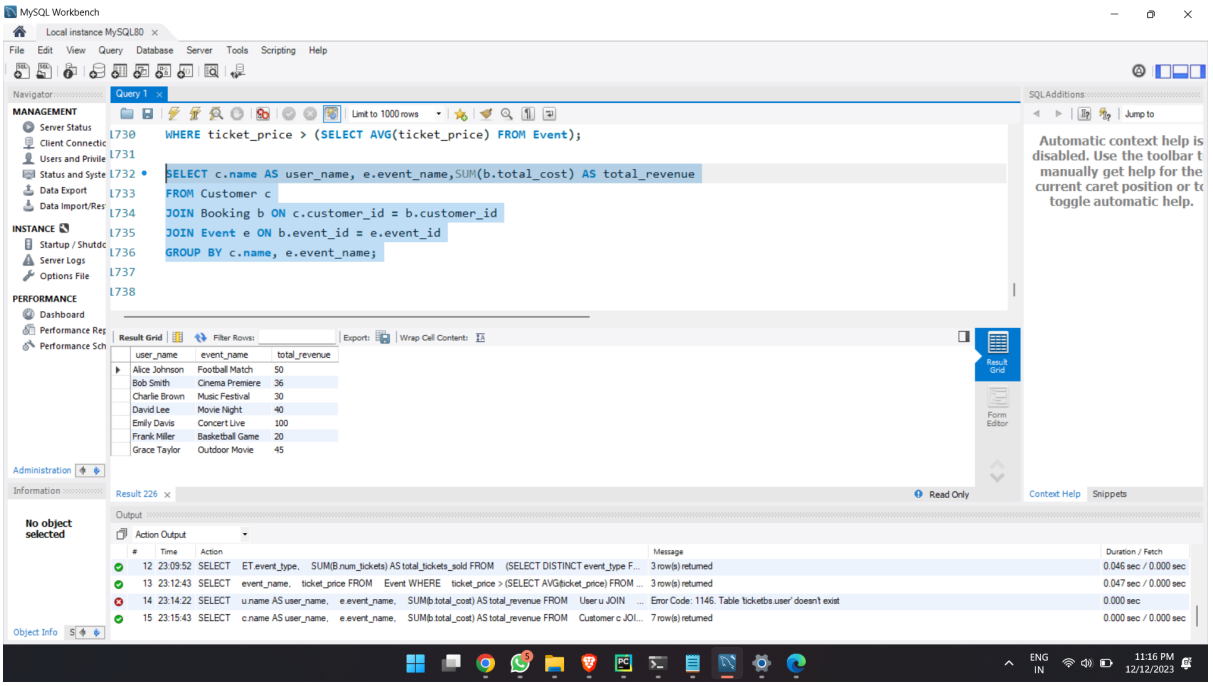
Q6. Calculate the Total Number of Tickets Sold for Each Event Type Using a Subquery in the FROM Clause.



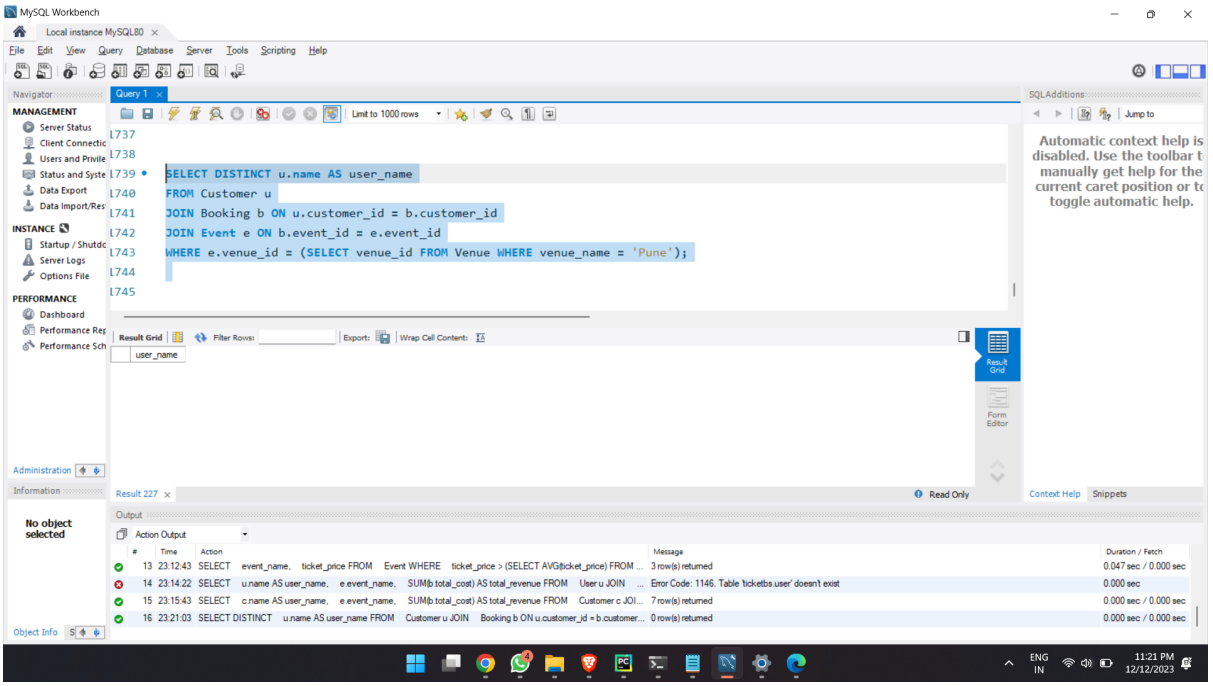
Q7. Find Events with Ticket Prices Higher Than the Average Ticket Price Using a Subquery in the WHERE Clause.



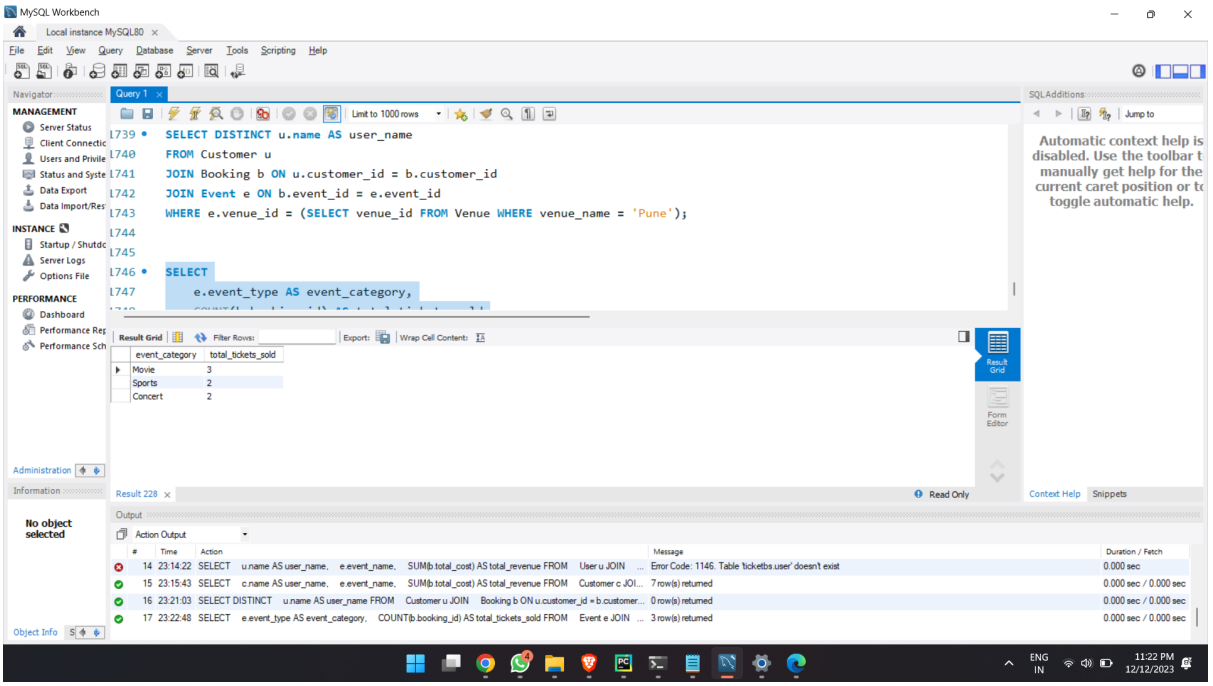
Q8. Calculate the Total Revenue Generated by Events for Each User Using a Correlated Subquery.



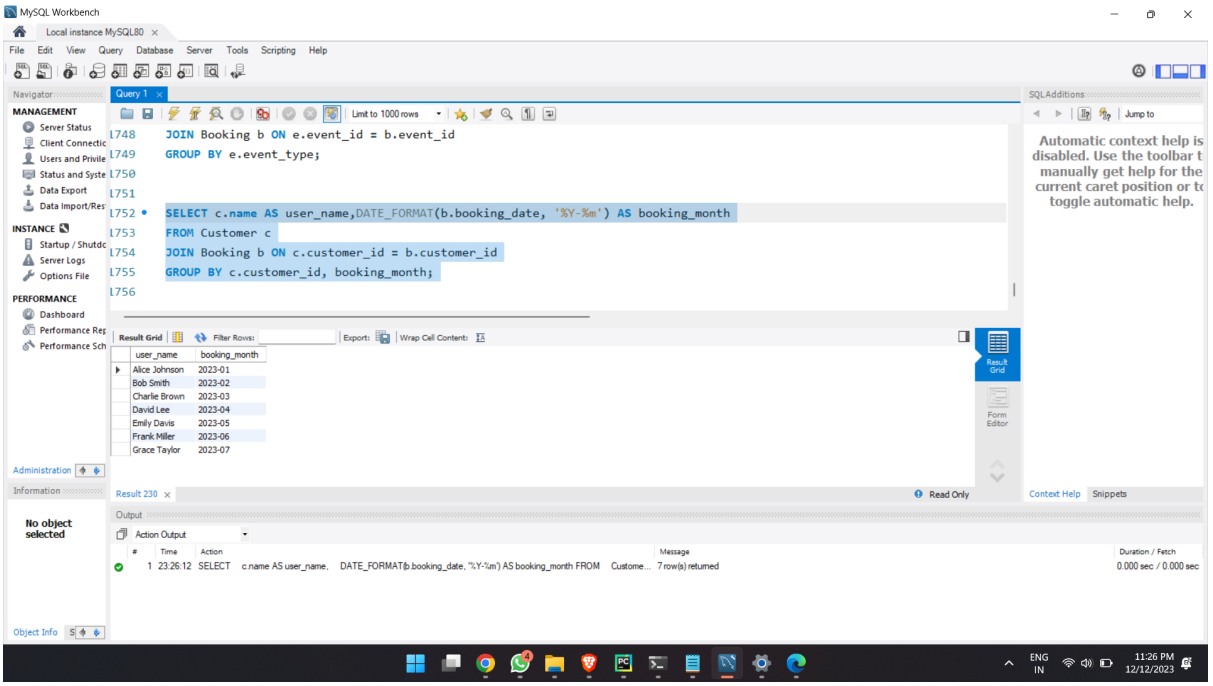
Q9. List Users Who Have Booked Tickets for Events in a Given Venue Using a Subquery in the WHERE Clause.



Q10. Calculate the Total Number of Tickets Sold for Each Event Category Using a Subquery with GROUP BY.



Q11. Find Users Who Have Booked Tickets for Events in each Month Using a Subquery with DATE_FORMAT.



Q12. Calculate the Average Ticket Price for Events in Each Venue Using a Subquery