|  |  |
| --- | --- |
| **Ex. No: 04** | **JOINS AND SET OPERATIONS** |
| Date | 06-02-24 |

**Objective:**

To execute the given queries using set operators and joins.

**Description:**

Set Operators

The three *set operators* union, intersect and minus allow to serially combine more than one select statements. Although more than one select statement will then be present, only *one* result set is then returned. The following list briefly describes the three set operations supported by Oracle SQL:

1) UNION

union all is very similar to union, however, it dismisses duplicate rows found across different select statements:

select col\_1, col\_2, col\_3 from table\_1 union

select col\_1, col\_2, col\_3 from table\_2;

2) INTERSECT

intersect only returns the rows that are found in all select statements:

select col\_1, col\_2, col\_3 from table\_1 intersect

select col\_1, col\_2, col\_3 from table\_2;

3) MINUS

minus returns all rows from the first select statements except those who are duplicated in a following select statement:

select col\_1, col\_2, col\_3 from table\_1 minus

select col\_1, col\_2, col\_3 from table\_2;

SQL JOIN

The JOIN keyword is used in an SQL statement to query data from two or more tables, based on a relationship between certain columns in these tables.

Tables in a database are often related to each other with keys.

Different SQL JOINs

Before we continue with examples, we will list the types of JOIN you can use, and the differences between them.

JOIN: Return rows when there is at least one match in both tables

LEFT JOIN: Return all rows from the left table, even if there are no matches in the right table

RIGHT JOIN: Return all rows from the right table, even if there are no matches in the left table

FULL JOIN: Return rows when there is a match in one of the tables

SQL INNER JOIN Keyword

The INNER JOIN keyword return rows when there is at least one match in both tables.

Syntax

|  |
| --- |
| SELEC column\_name(s) FROM table\_name1 INNER JOIN table\_name2 ON table\_name1.column\_name=table\_name2.column\_name |

PS: INNER JOIN is the same as JOIN.

SQL LEFT JOIN Keyword

The LEFT JOIN keyword returns all rows from the left table (table\_name1), even if there are no matches in the right table (table\_name2).

Syntax

|  |
| --- |
| SELECT column\_name(s) FROM table\_name1 LEFT OUTER JOIN table\_name2 ON table\_name1.column\_name=table\_name2.column\_name |

SQL RIGHT JOIN Keyword

The RIGHT JOIN keyword Return all rows from the right table (table\_name2), even if there are no matches in the left table (table\_name1).

Syntax

|  |
| --- |
| SELECT column\_name(s) FROM table\_name1 RIGHT OUTER JOIN table\_name2 ON table\_name1.column\_name=table\_name2.column\_name |

SQL FULL JOIN Keyword

The FULL JOIN keyword return rows when there is a match in one of the tables.

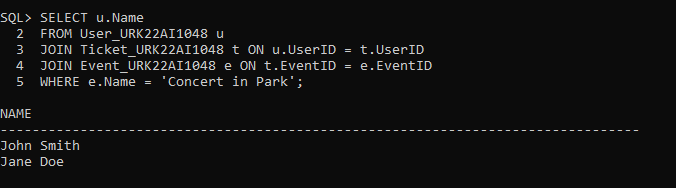
Syntax

|  |
| --- |
| SELECT column\_name(s) FROM table\_name1 FULL OUTER JOIN table\_name2 ON table\_name1.column\_name=table\_name2.column\_name |

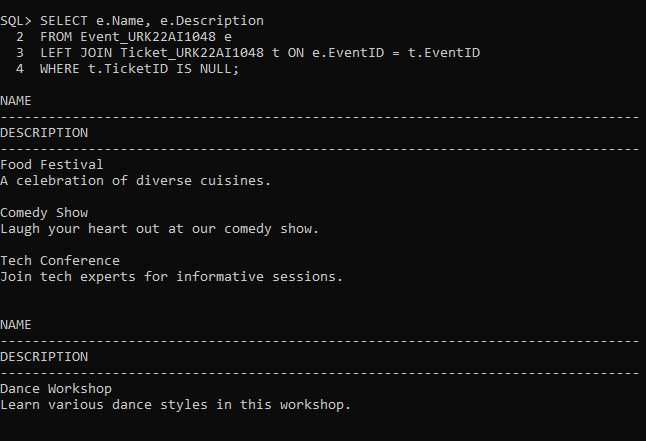
The JOIN keyword is used in an SQL statement to query data from two or more tables, based on a relationship between certain columns in these tables. Whenever a query is written which refers more than one table that needs the help of joins.

**Questions:**

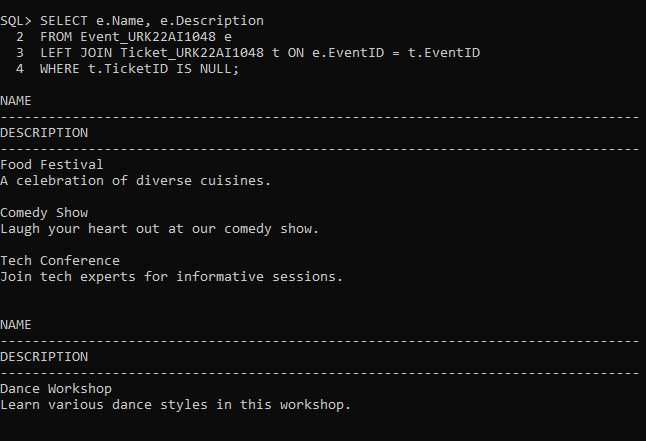
1. Retrieve the names of users who have registered for the "Concert in Park" event:



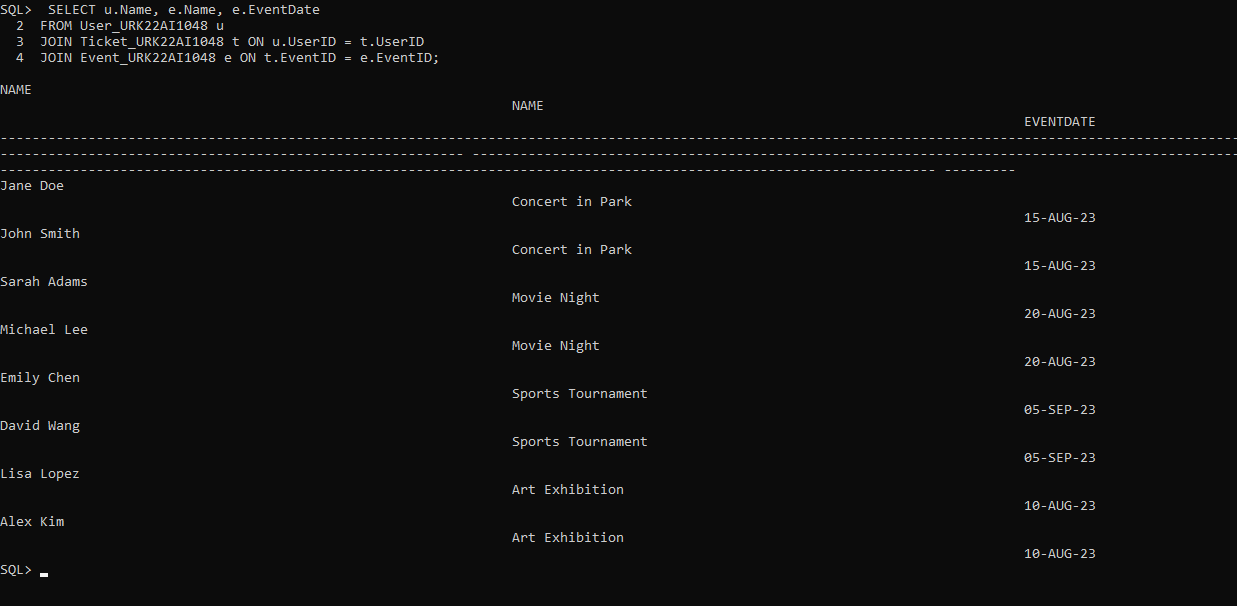
2. Find the details of events (name, date, and time) that Sarah Adams has registered for.



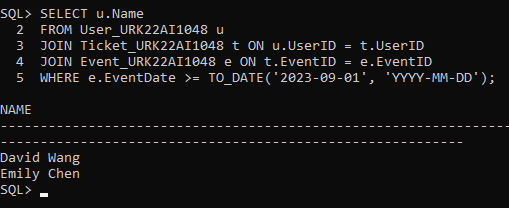
3. List the events (name and description) that do not have any registered participants.



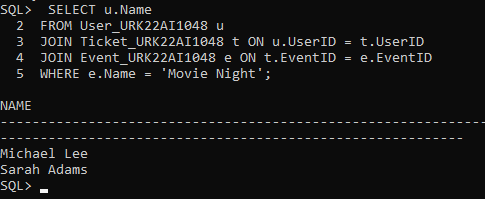
4. Retrieve the names of users and the events they have registered for, along with the event dates.



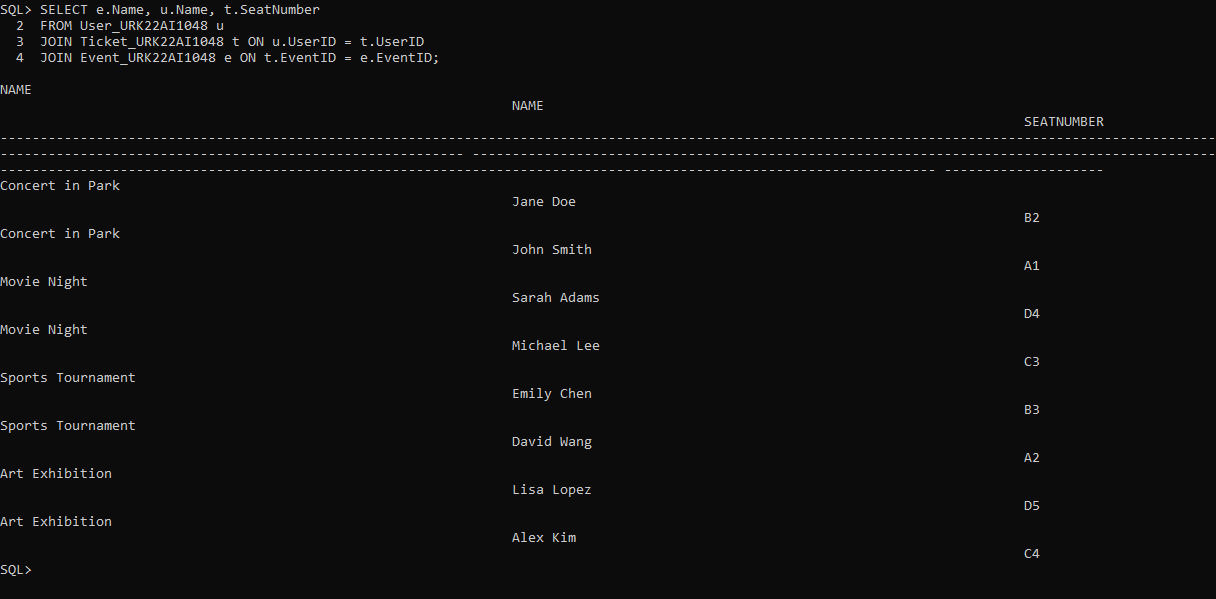
5. Find the names of users who have registered for events taking place on or after September 1, 2023.



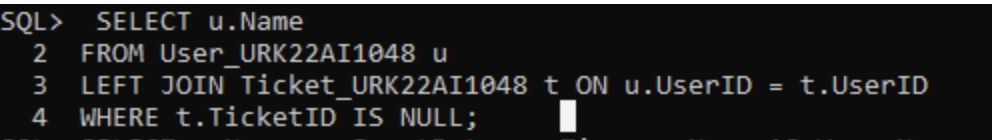
6. Retrieve the names of users who have booked tickets for the "Movie Night" event.



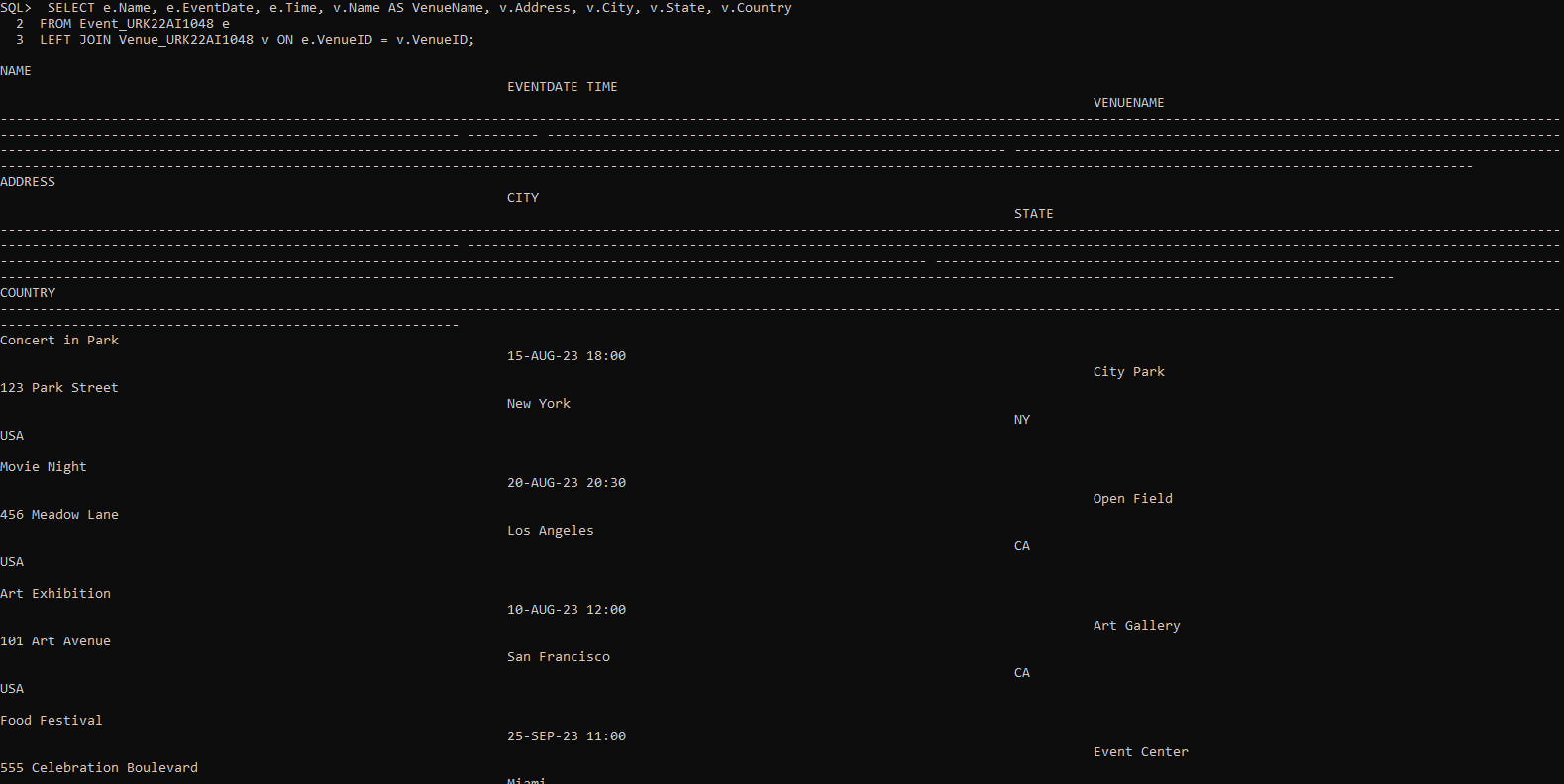
7. List the event names, user names, and seat numbers for all booked tickets.



8. Find the names of users who have not booked any tickets for any event.



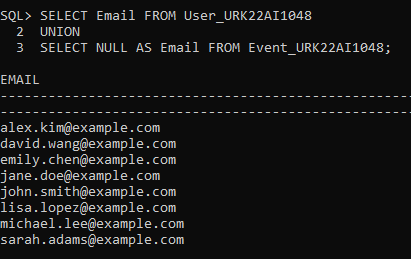
9. Perform Left Outer Join to Retrieve Event Details along with Venue Information.



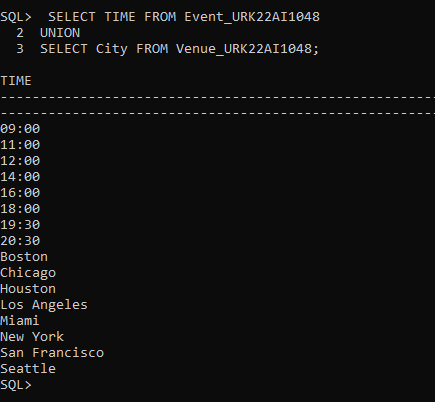
10. Perform Full Outer Join to Retrieve Combined Event and Venue Information.



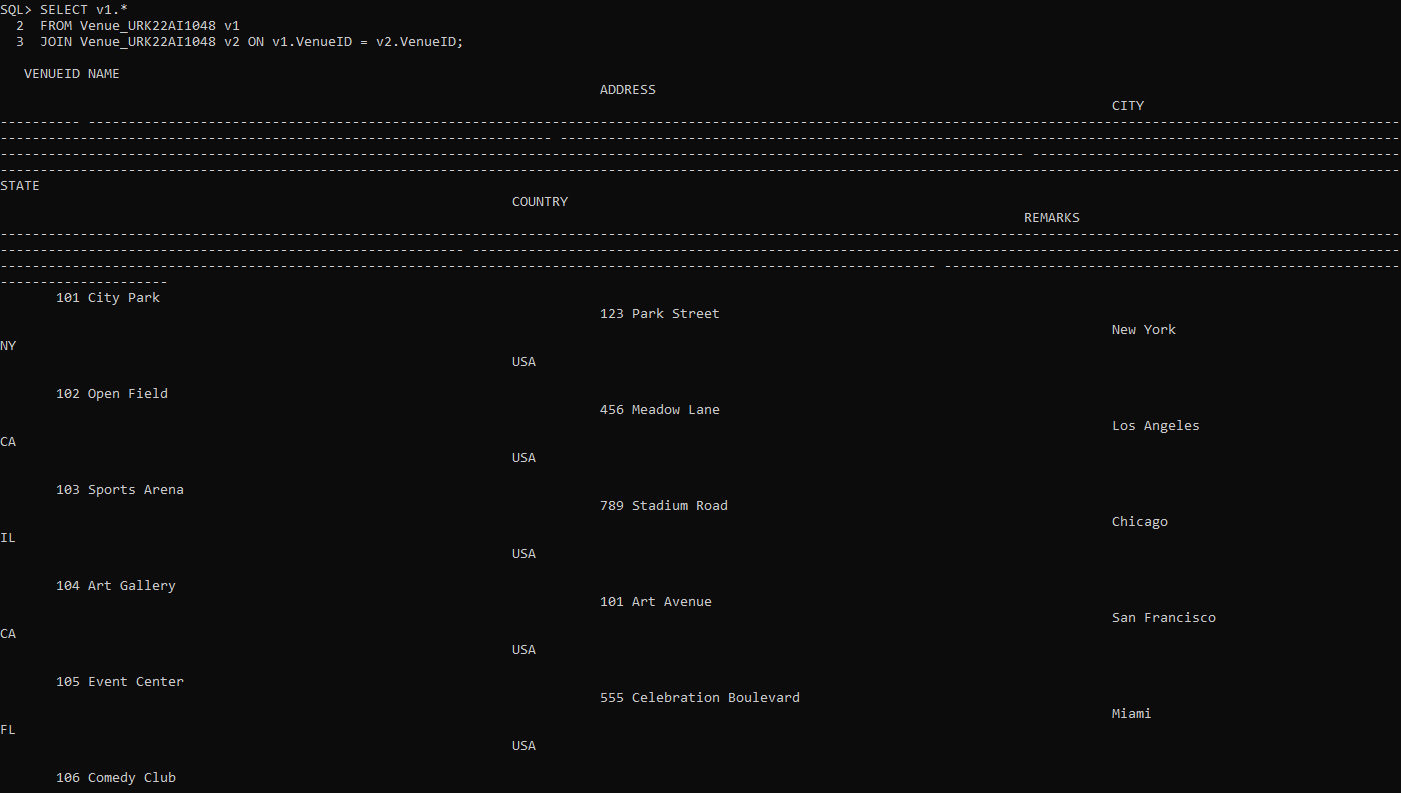
11. Write a query to retrieve a list of unique email addresses from both the User and Event tables using set operations.



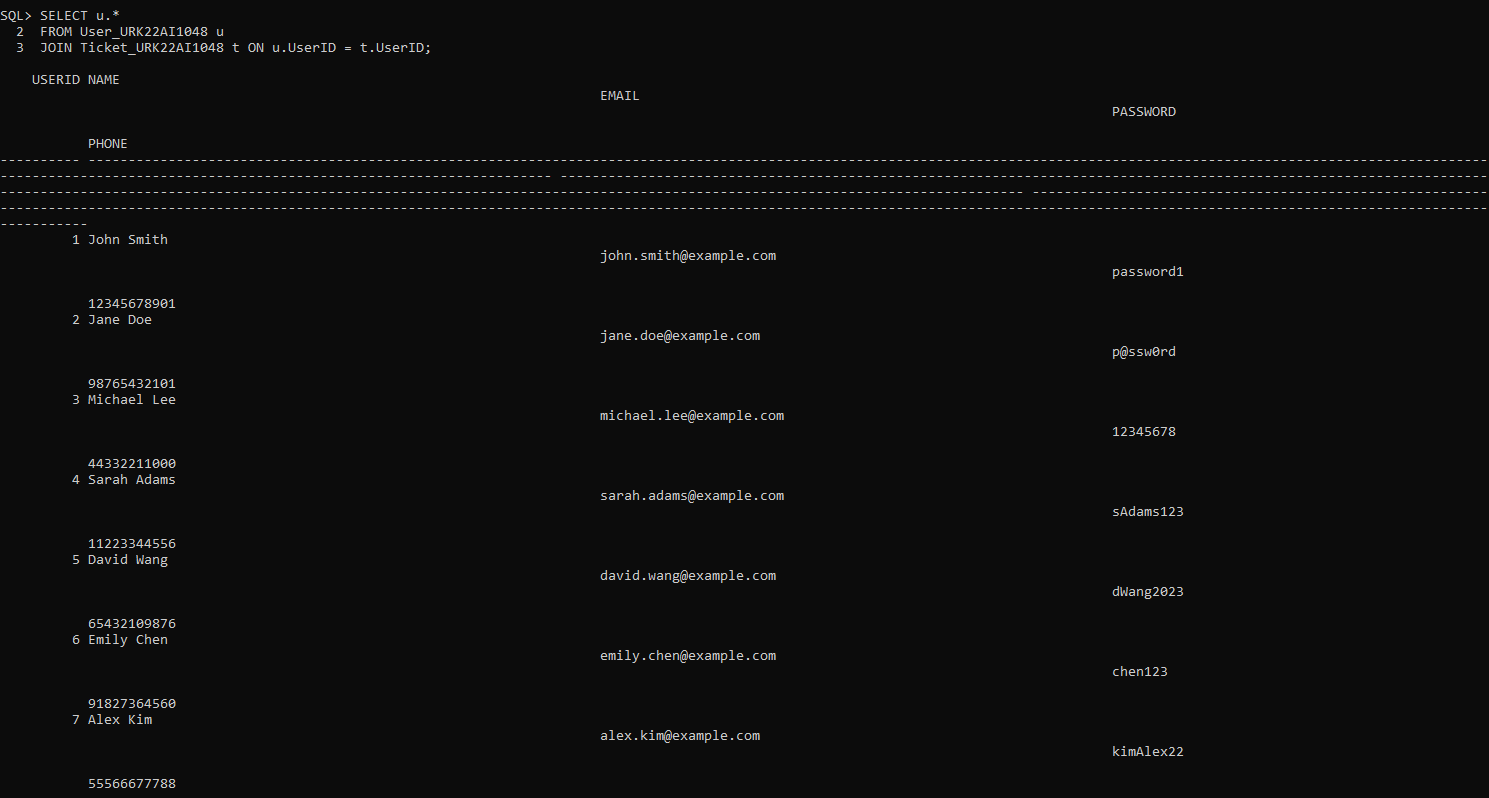
12.Write a query to retrieve a list of unique cities where events are scheduled or venues are located.



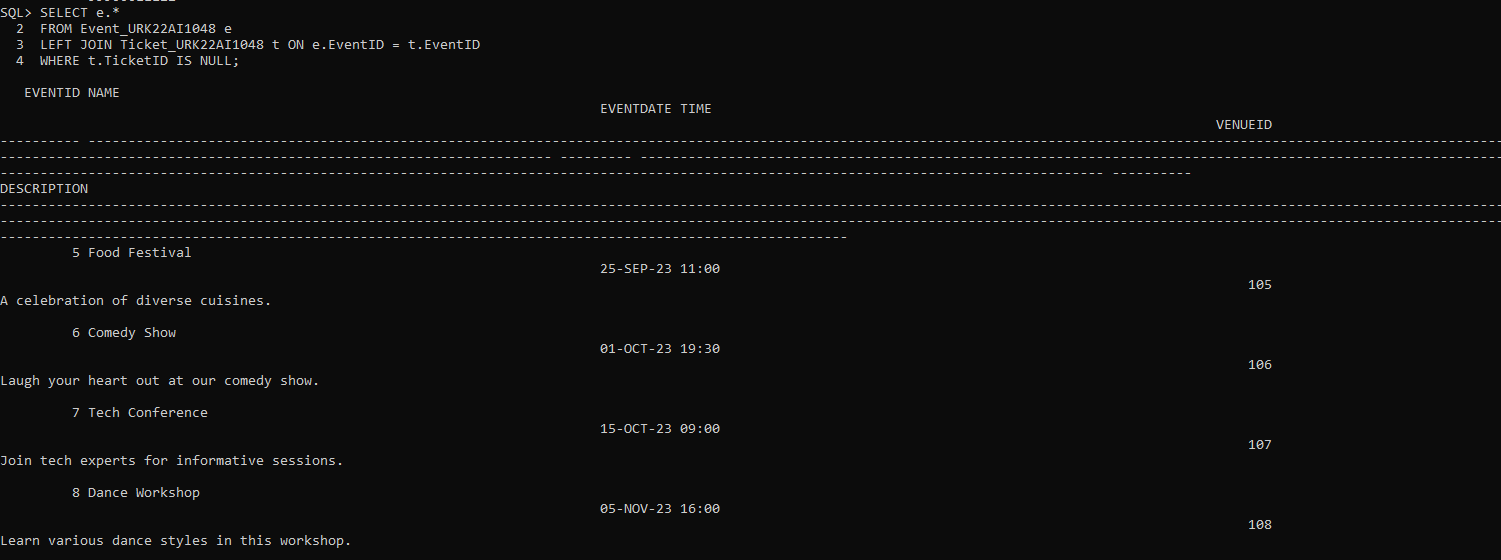
13.Write a query to display the details of Venue ID conducted in the same Venues.



14. Write a query to display the details of User ID who are users and have registered for an event.



15. Write a query to display the details of Event ID which are events but not booked by any one.



**Result:**

The given queries executed by the set operations and joins successfully.