**DBT Assignment 1**

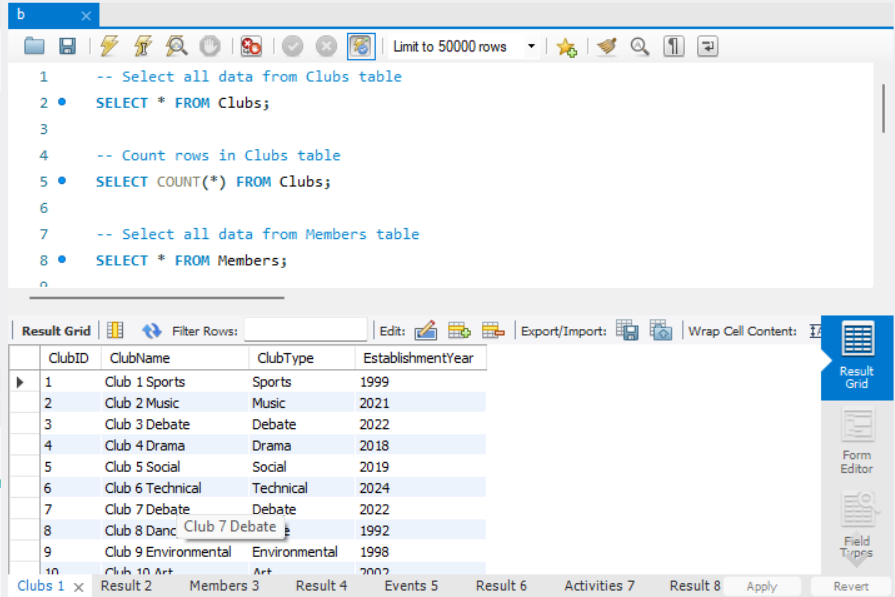
Name : Mohul Y P

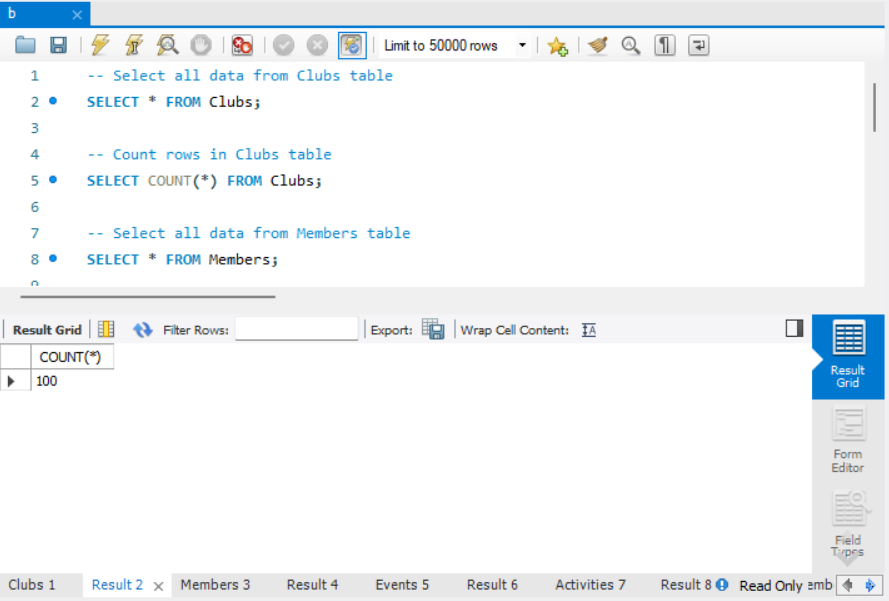
SRN : PES1UG22CS360

b)**select and count queries on “clubs” table**

**SELECT \* FROM Clubs;**

**SELECT COUNT(\*) FROM Clubs;**

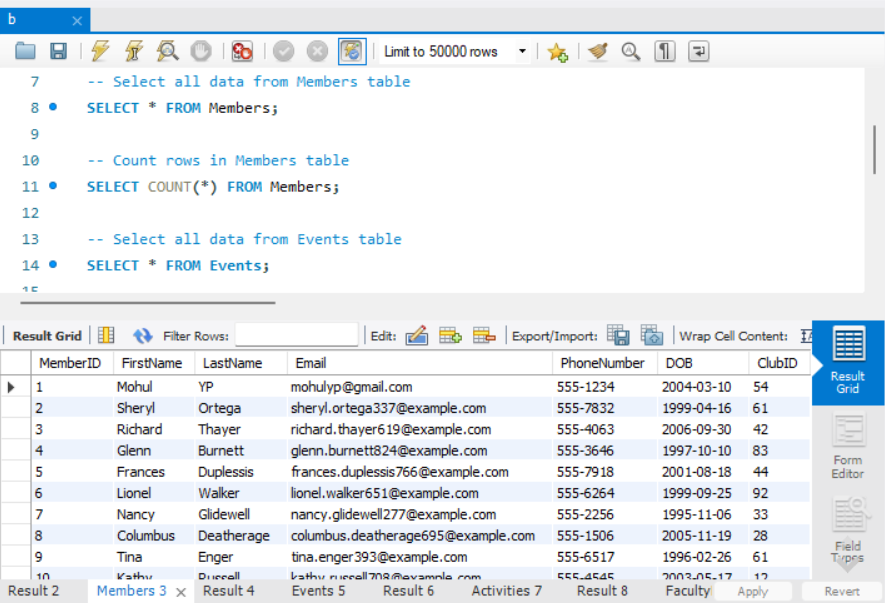


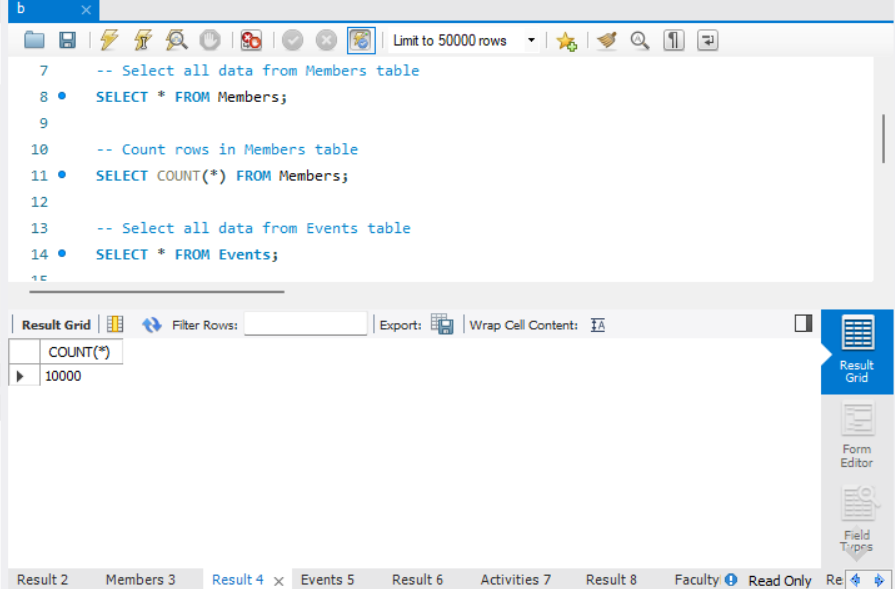


**select and count queries on “member” table**

**SELECT \* FROM Members;**

**SELECT COUNT(\*) FROM Members;**

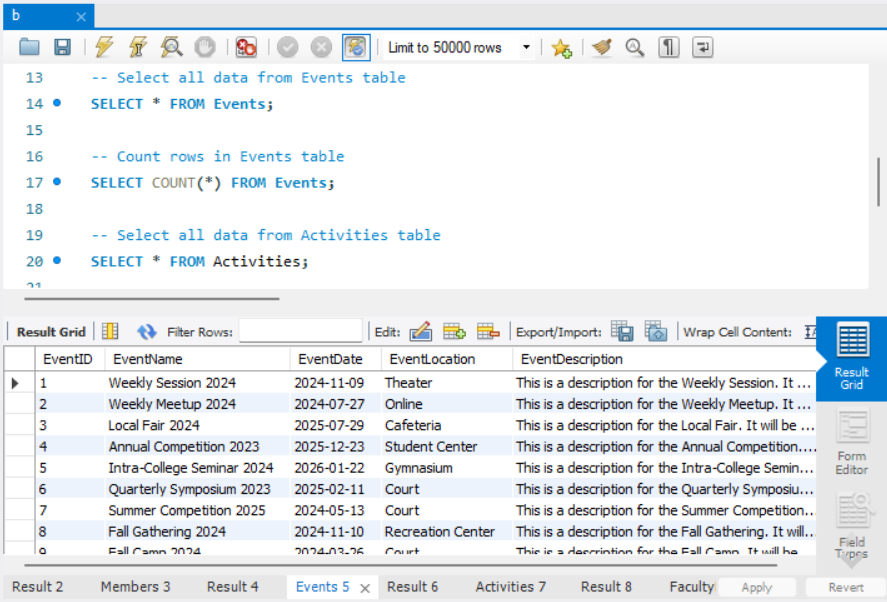


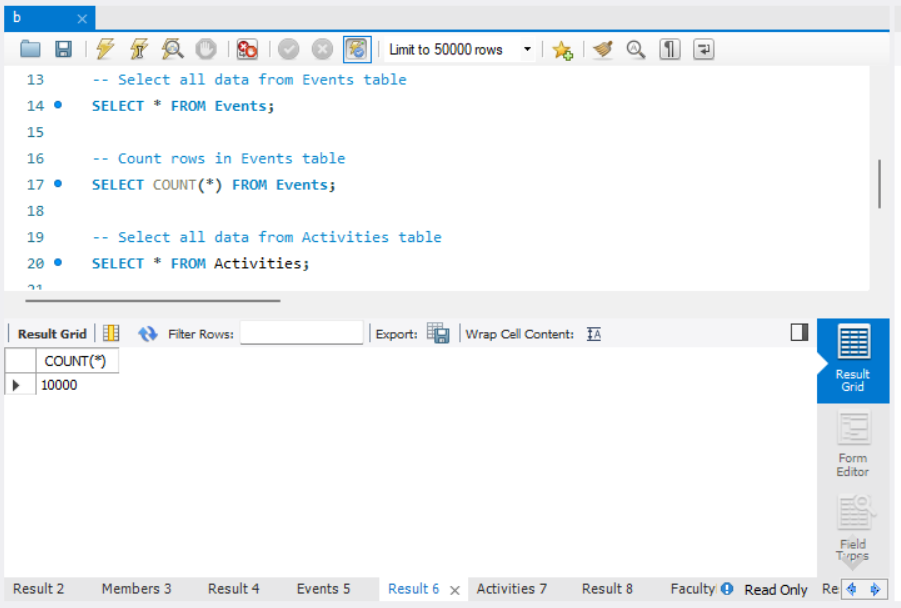


**select and count queries on “events” table**

**SELECT \* FROM Events;**

**SELECT COUNT(\*) FROM Events;**

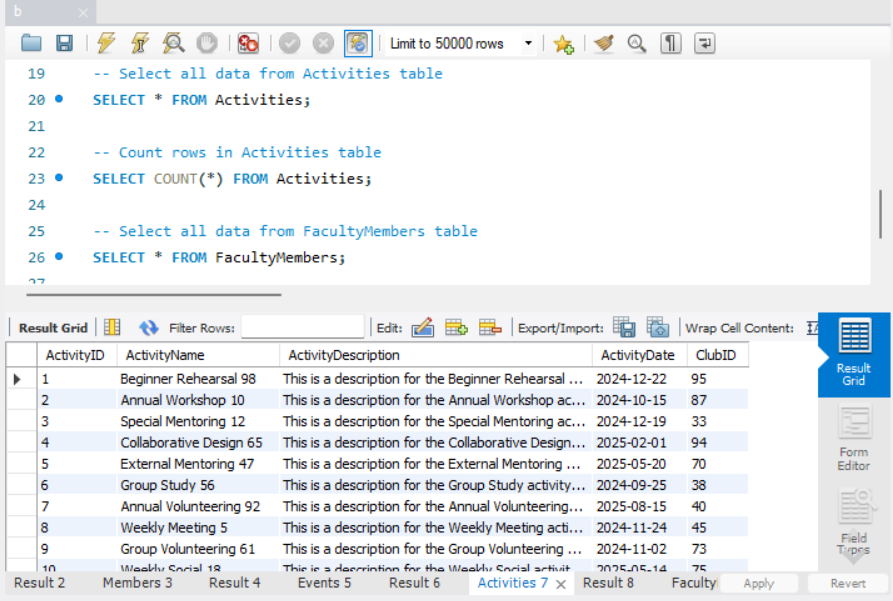


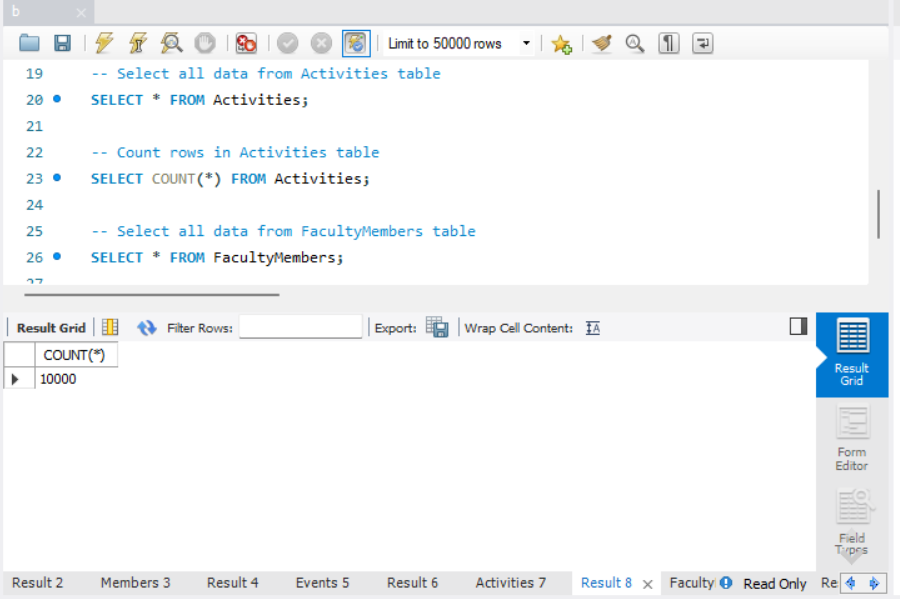


**select and count queries on “activities” table**

**SELECT \* FROM Activities;**

**SELECT COUNT(\*) FROM Activities;**

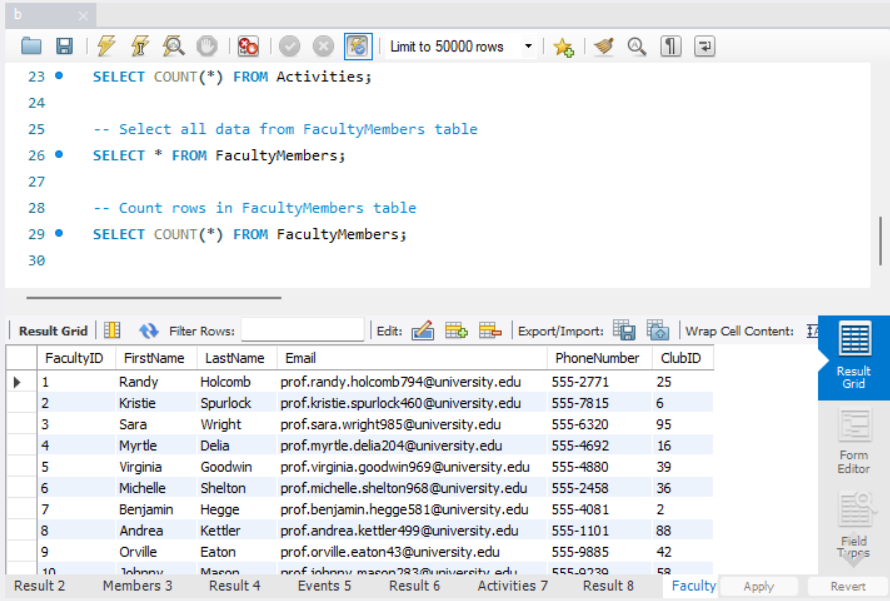


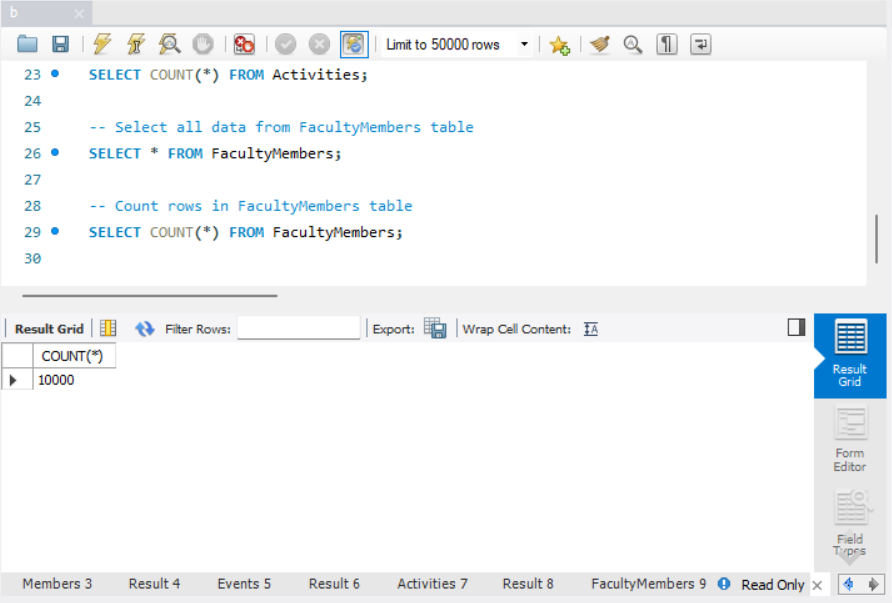


**select and count queries on “faculty members” table**

**SELECT \* FROM FacultyMembers;**

**SELECT COUNT(\*) FROM FacultyMembers;**





**INDEX SCAN**

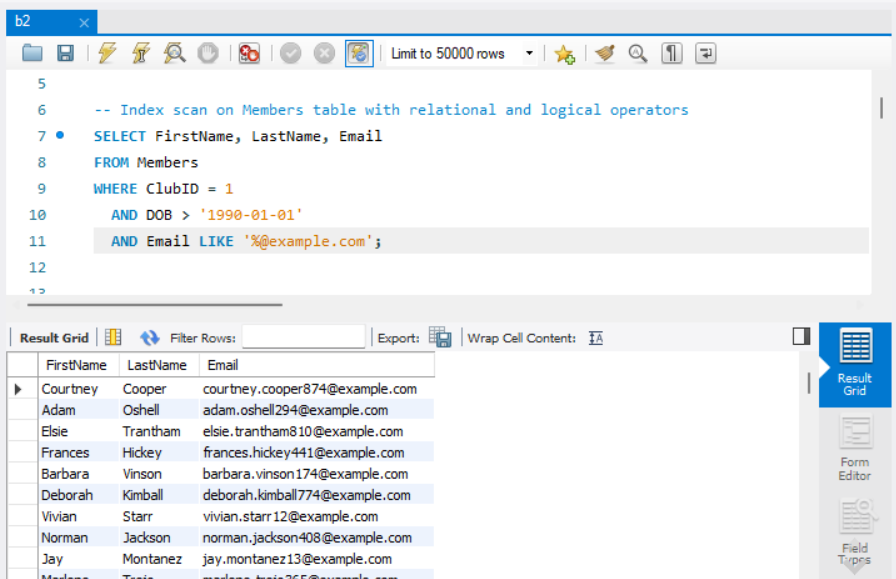
**1)** **SELECT FirstName, LastName, Email**

**FROM Members**

**WHERE ClubID = 1**

**AND DOB > '1990-01-01'**

**AND Email LIKE '%@example.com';**

****

**TABLE SCAN**

1. **– table scan on members to find members born after 1990 ordered by first name who are not in sports clubs established after 2000**

**SELECT \***

**FROM Members m**

**WHERE m.ClubID NOT IN (**

**SELECT c.ClubID**

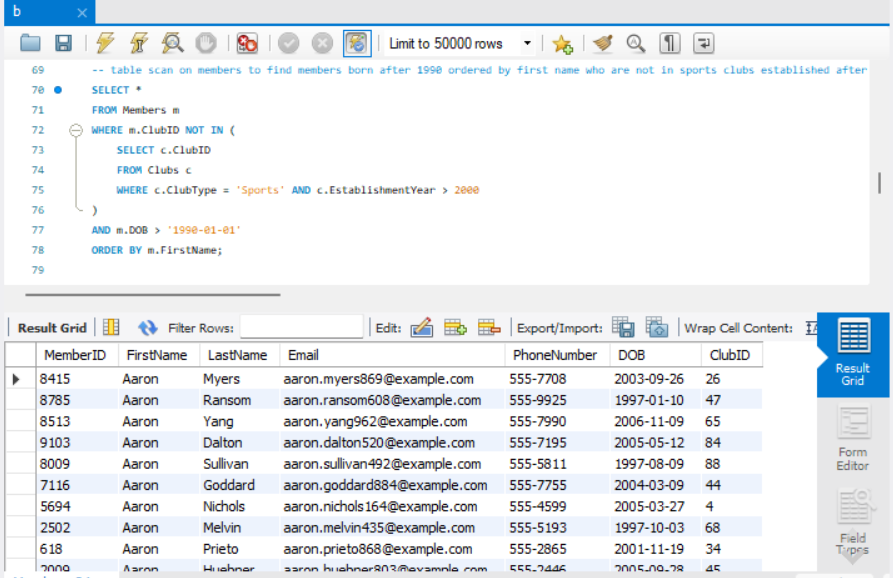
**FROM Clubs c**

**WHERE c.ClubType = 'Sports' AND c.EstablishmentYear > 2000**

**)**

**AND m.DOB > '1990-01-01'**

**ORDER BY m.FirstName;**

****

**2) --table scan on join of activities and members table.**

**SELECT a.ActivityName, COUNT(m.MemberID) AS MemberCount**

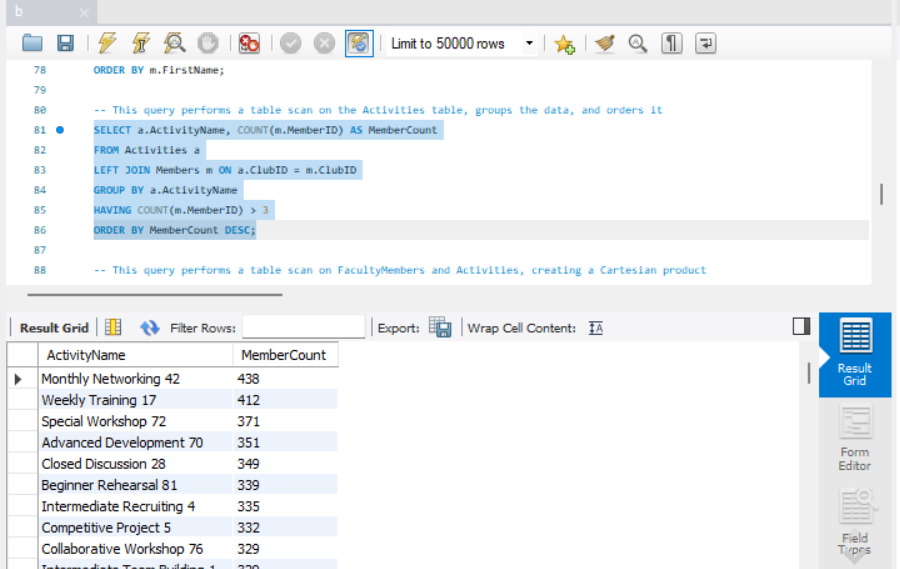
**FROM Activities a**

**LEFT JOIN Members m ON a.ClubID = m.ClubID**

**GROUP BY a.ActivityName**

**HAVING COUNT(m.MemberID) > 3**

**ORDER BY MemberCount DESC;**

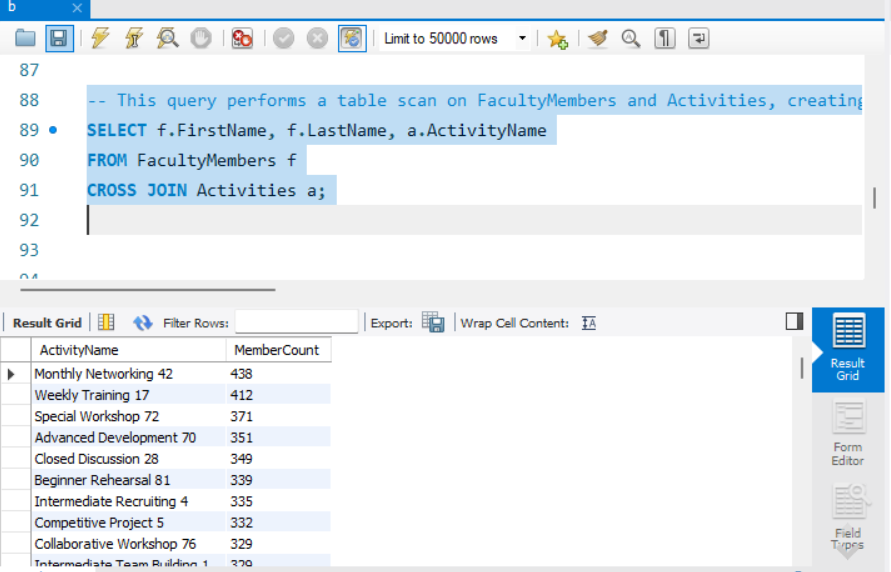
****

**3)** **-- This query performs a table scan on FacultyMembers and Activities, creating a Cartesian product**

**SELECT f.FirstName, f.LastName, a.ActivityName**

**FROM FacultyMembers f**

**CROSS JOIN Activities a;**

****

**queries with multi-table joins involving 3 tables; including both "SELECT \*" and**

**conditional "SELECT" queries with a subset of columns.**

**1)** **SELECT c.ClubName, COUNT(m.MemberID) AS MemberCount**

**FROM Members m**

**INNER JOIN Clubs c ON m.ClubID = c.ClubID**

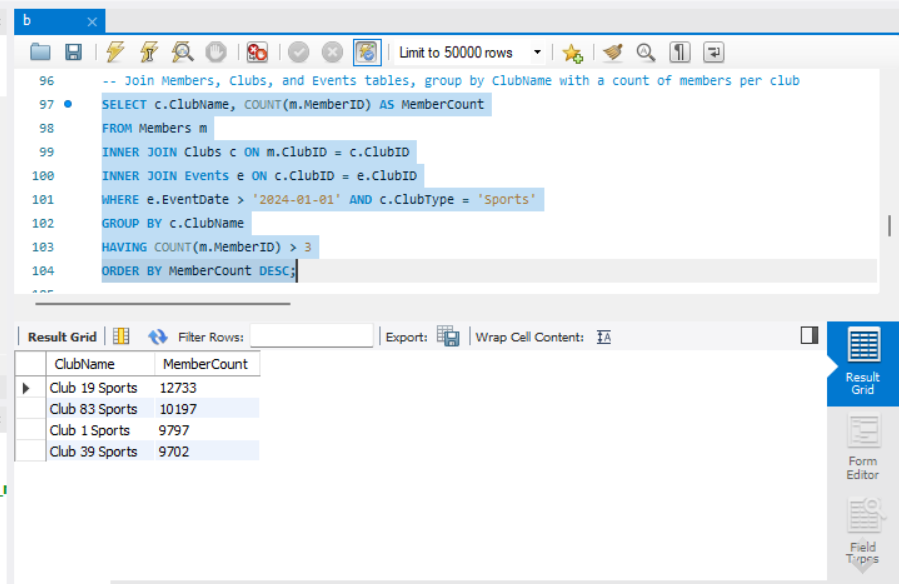
**INNER JOIN Events e ON c.ClubID = e.ClubID**

**WHERE e.EventDate > '2024-01-01' AND c.ClubType = 'Sports'**

**GROUP BY c.ClubName**

**HAVING COUNT(m.MemberID) > 3**

**ORDER BY MemberCount DESC;**

****

**2)**

**SELECT \***

**FROM Members m**

**INNER JOIN Clubs c ON m.ClubID = c.ClubID**

**INNER JOIN Events e ON c.ClubID = e.ClubID**

**WHERE e.EventDate IN (**

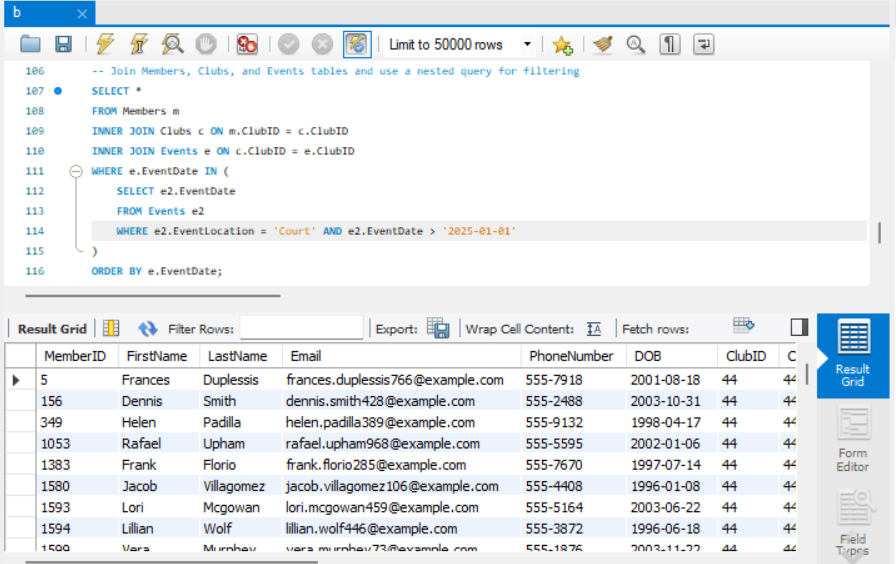
**SELECT e2.EventDate**

**FROM Events e2**

**WHERE e2.EventLocation = 'Hall A' AND e2.EventDate > '2025-01-01'**

**)**

**ORDER BY e.EventDate;**

****

**3)** **CREATE VIEW ClubMembers AS**

**SELECT c.ClubName, m.FirstName, m.LastName, m.Email**

**FROM Clubs c**

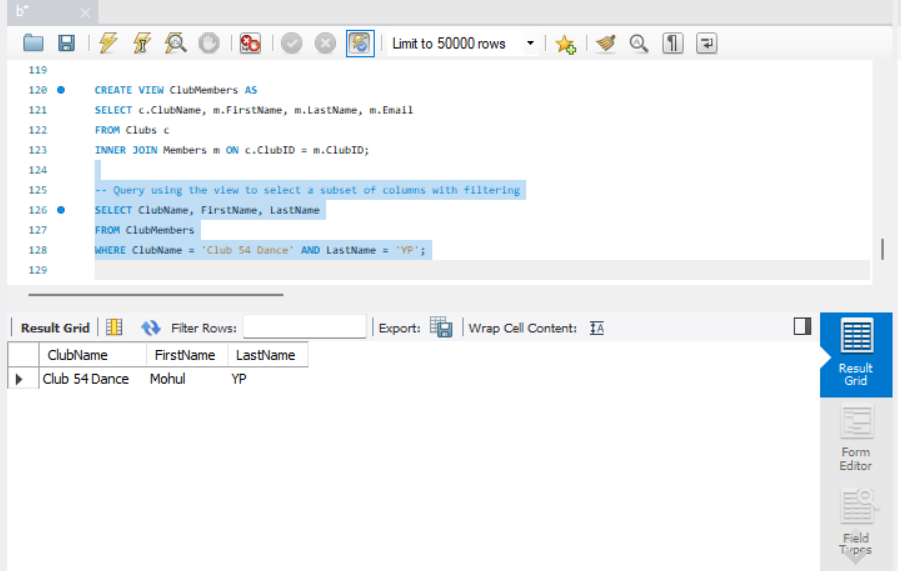
**INNER JOIN Members m ON c.ClubID = m.ClubID;**

**-- Query using the view to select a subset of columns with filtering**

**SELECT ClubName, FirstName, LastName**

**FROM ClubMembers**

**WHERE ClubName = 'Club 54 Dance' AND LastName = 'YP';**

****

**EXPLAIN ANALYZE on each of the queries mentioned above :**

**INDEX SCAN**

**1)** **EXPLAIN ANALYZE**

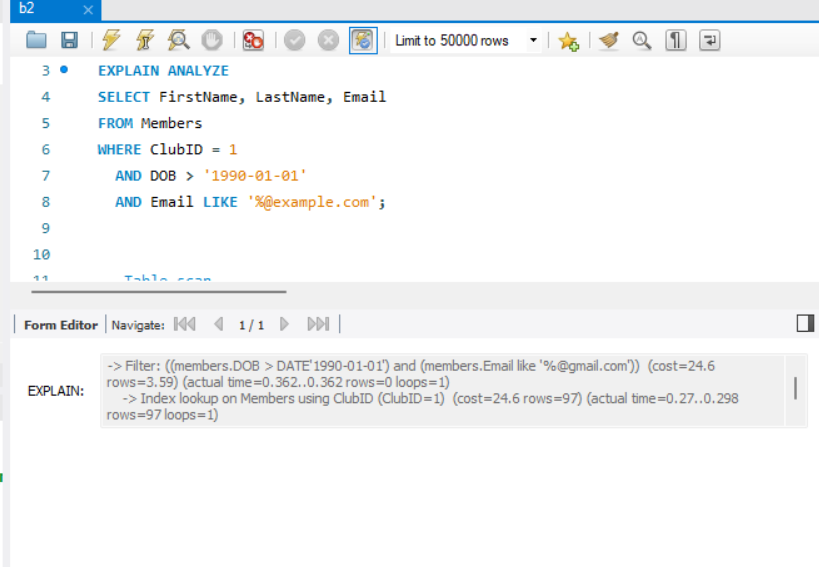
**SELECT FirstName, LastName, Email**

**FROM Members**

**WHERE ClubID = 1**

**AND DOB > '1990-01-01'**

**AND Email LIKE '%@example.com';**

****

**TABLE SCAN**

**1)**

**EXPLAIN ANALYZE**

**SELECT \***

**FROM Members m**

**WHERE m.ClubID NOT IN (**

**SELECT c.ClubID**

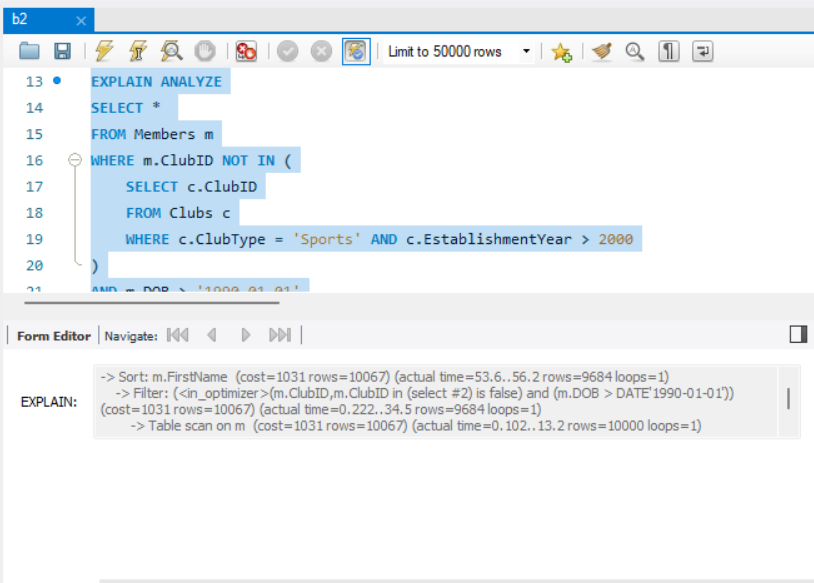
**FROM Clubs c**

**WHERE c.ClubType = 'Sports' AND c.EstablishmentYear > 2000**

**)**

**AND m.DOB > '1990-01-01'**

**ORDER BY m.FirstName;**

****

**2)**

**EXPLAIN ANALYZE**

**SELECT a.ActivityName, COUNT(m.MemberID) AS MemberCount**

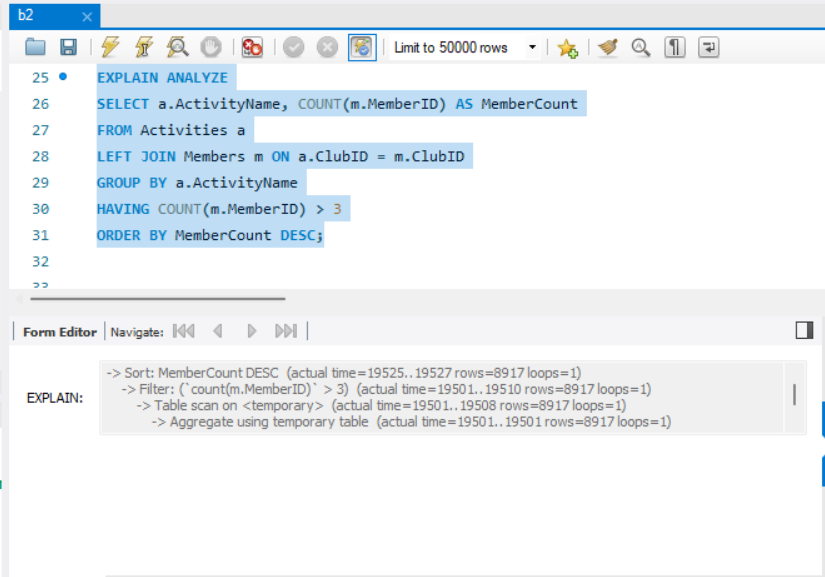
**FROM Activities a**

**LEFT JOIN Members m ON a.ClubID = m.ClubID**

**GROUP BY a.ActivityName**

**HAVING COUNT(m.MemberID) > 3**

**ORDER BY MemberCount DESC;**

****

**3)**

**EXPLAIN ANALYZE**

**SELECT f.FirstName, f.LastName, a.ActivityName**

**FROM FacultyMembers f**

**CROSS JOIN Activities a;**

**queries with multi-table joins involving 3 tables; including both "SELECT \*" and**

**conditional "SELECT" queries with a subset of columns.**

**1)**

**EXPLAIN ANALYZE**

**SELECT c.ClubName, COUNT(m.MemberID) AS MemberCount**

**FROM Members m**

**INNER JOIN Clubs c ON m.ClubID = c.ClubID**

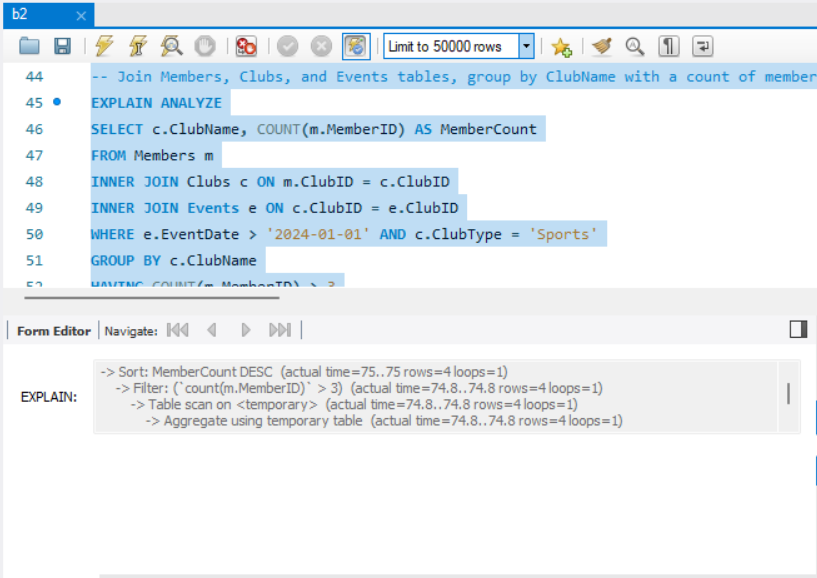
**INNER JOIN Events e ON c.ClubID = e.ClubID**

**WHERE e.EventDate > '2024-01-01' AND c.ClubType = 'Sports'**

**GROUP BY c.ClubName**

**HAVING COUNT(m.MemberID) > 3**

**ORDER BY MemberCount DESC;**

****

**2)**

**EXPLAIN ANALYZE**

**SELECT \***

**FROM Members m**

**INNER JOIN Clubs c ON m.ClubID = c.ClubID**

**INNER JOIN Events e ON c.ClubID = e.ClubID**

**WHERE e.EventDate IN (**

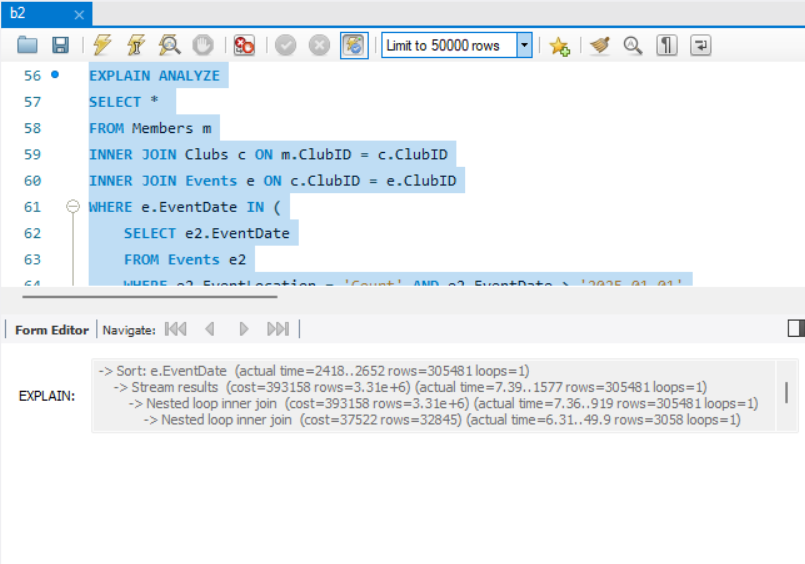
**SELECT e2.EventDate**

**FROM Events e2**

**WHERE e2.EventLocation = 'Court' AND e2.EventDate > '2025-01-01'**

**)**

**ORDER BY e.EventDate;**

****

**3)**

**CREATE VIEW ClubMembers AS**

**SELECT c.ClubName, m.FirstName, m.LastName, m.Email**

**FROM Clubs c**

**INNER JOIN Members m ON c.ClubID = m.ClubID;**

**EXPLAIN ANALYZE**

**-- Query using the view to select a subset of columns with filtering**

**SELECT ClubName, FirstName, LastName**

**FROM ClubMembers**

**WHERE ClubName = 'Club 54 Dance' AND LastName = 'YP';**

****

**c)**

**Indexes made to improve performance :**

**CREATE INDEX idx\_members\_dob\_firstname ON Members(DOB, FirstName);**

**CREATE INDEX idx\_clubs\_clubtype\_establishmentyear ON Clubs(ClubType, EstablishmentYear);**

**CREATE INDEX idx\_clubs\_clubname ON Clubs(ClubName);**

**CREATE INDEX idx\_events\_eventdate ON Events(EventDate);**

**1)analyzing below query :**

**SELECT \***

**FROM Members m**

**WHERE m.ClubID NOT IN (**

**SELECT c.ClubID**

**FROM Clubs c**

**WHERE c.ClubType = 'Sports' AND c.EstablishmentYear > 2000**

**)**

**AND m.DOB > '1990-01-01'**

**ORDER BY m.FirstName;**

****

**2)analyzing below query :**

**SELECT c.ClubName, COUNT(m.MemberID) AS MemberCount**

**FROM Members m**

**INNER JOIN Clubs c ON m.ClubID = c.ClubID**

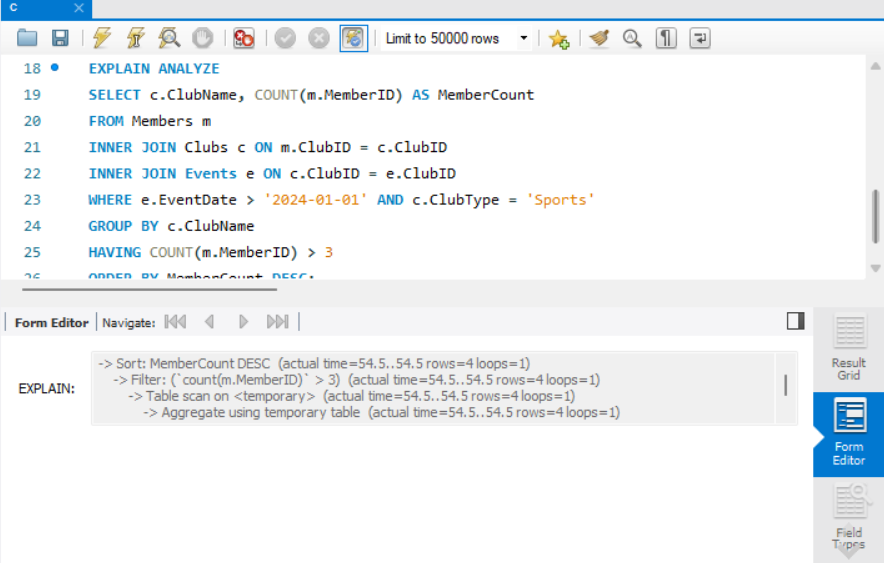
**INNER JOIN Events e ON c.ClubID = e.ClubID**

**WHERE e.EventDate > '2024-01-01' AND c.ClubType = 'Sports'**

**GROUP BY c.ClubName**

**HAVING COUNT(m.MemberID) > 3**

**ORDER BY MemberCount DESC;**

****

**d) The multi join query to optimize :**

**1)**

**SELECT c.ClubName, COUNT(m.MemberID) AS MemberCount**

**FROM Members m**

**INNER JOIN Clubs c ON m.ClubID = c.ClubID**

**INNER JOIN Events e ON c.ClubID = e.ClubID**

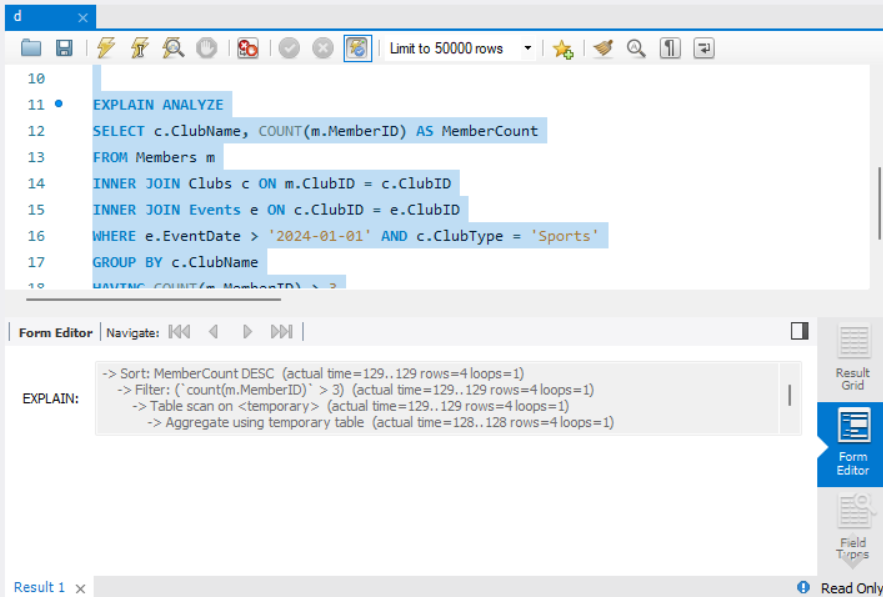
**WHERE e.EventDate > '2024-01-01' AND c.ClubType = 'Sports'**

**GROUP BY c.ClubName**

**HAVING COUNT(m.MemberID) > 3**

**ORDER BY MemberCount DESC;**

**Running EXPLAIN ANALYZE on the above query :**

****

**Join order variation 1 : Start with clubs**

**SELECT c.ClubName, COUNT(m.MemberID) AS MemberCount**

**FROM Clubs c**

**INNER JOIN Events e ON c.ClubID = e.ClubID**

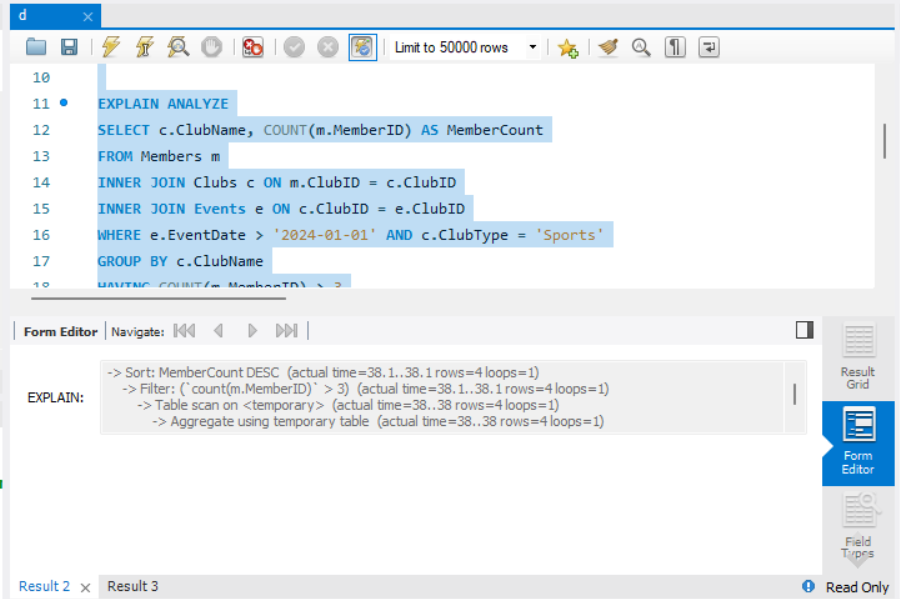
**INNER JOIN Members m ON c.ClubID = m.ClubID**

**WHERE e.EventDate > '2024-01-01' AND c.ClubType = 'Sports'**

**GROUP BY c.ClubName**

**HAVING COUNT(m.MemberID) > 3**

**ORDER BY MemberCount DESC;**

****

**Join order variation 2 : Start with events**

**SELECT c.ClubName, COUNT(m.MemberID) AS MemberCount**

**FROM Events e**

**INNER JOIN Clubs c ON e.ClubID = c.ClubID**

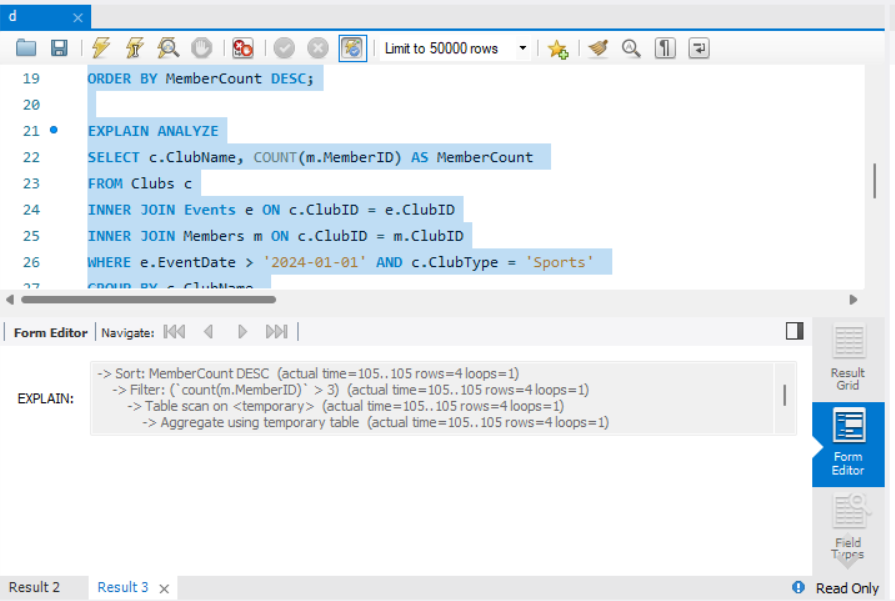
**INNER JOIN Members m ON c.ClubID = m.ClubID**

**WHERE e.EventDate > '2024-01-01' AND c.ClubType = 'Sports'**

**GROUP BY c.ClubName**

**HAVING COUNT(m.MemberID) > 3**

**ORDER BY MemberCount DESC;**

****

**Incorporating a variety of join types such as outer joins, subqueries, etc., to diversify optimization approaches.**

**1)** **Using LEFT JOIN with Subquery Filtering**

**SELECT c.ClubName, COUNT(m.MemberID) AS MemberCount**

**FROM Clubs c**

**LEFT JOIN Members m ON c.ClubID = m.ClubID**

**WHERE c.ClubType = 'Sports'**

**AND c.ClubID IN (**

**SELECT DISTINCT e.ClubID**

**FROM Events e**

**WHERE e.EventDate > '2024-01-01'**

**)**

**GROUP BY c.ClubName**

**HAVING COUNT(m.MemberID) > 3**

**ORDER BY MemberCount DESC;**

****

1. **Using Common Table Expressions with INNER JOIN**

**WITH FilteredClubs AS (**

**SELECT ClubID, ClubName**

**FROM Clubs**

**WHERE ClubType = 'Sports'**

**),**

**ActiveClubs AS (**

**SELECT DISTINCT fc.ClubID, fc.ClubName**

**FROM FilteredClubs fc**

**INNER JOIN Events e ON fc.ClubID = e.ClubID**

**WHERE e.EventDate > '2024-01-01'**

**)**

**SELECT ac.ClubName, COUNT(m.MemberID) AS MemberCount**

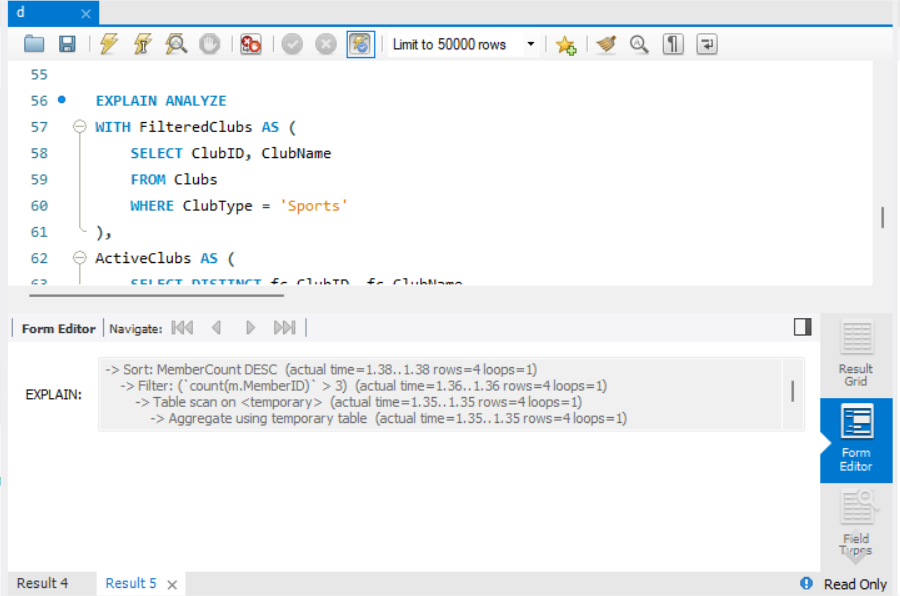
**FROM ActiveClubs ac**

**INNER JOIN Members m ON ac.ClubID = m.ClubID**

**GROUP BY ac.ClubName**

**HAVING COUNT(m.MemberID) > 3**

**ORDER BY MemberCount DESC;**

****

**e)**

**Part 1:**

**Parse tree:**

**Relational Algebric Expression**

**Initial query tree**

**Part 2:**