|  |  |
| --- | --- |
| **Project name** | **Gym Management System** |
| **Phase** | **03** |

**Write 6 questions on your case study and implement their solution with screenshots. 3 queries on joins (inner, left, right & full) and rest of them should be on nested/correlated queries. Each having 4 or more participating tables.**

**SUB QUERIES:**

1. **Display the name of those members who attend one class which is instructed by trainers who have specialization in ‘Body Building’.**

SELECT m.FirstName, m.LastName

FROM Member m

WHERE EXISTS (

SELECT \*

FROM Attendance AS a

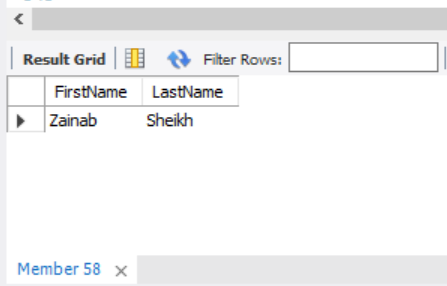
Join Class AS c on a.ClassID = c.ClassID

JOIN Trainer AS t ON c.TrainerID = t.TrainerID

WHERE a.MemberID = m.MemberID

AND t.Specialization = 'Bodybuilding'

);



1. **Display the names of all the member who attend the class which scheduled on 2024-6-15 and 2024-6-16**

SELECT FirstName, LastName

FROM Member

WHERE MemberID IN (

SELECT DISTINCT a.MemberID

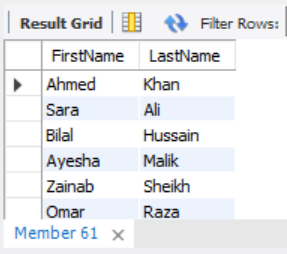
FROM Attendance AS a

join Class as c on a.ClassID = c.ClassID

JOIN Trainer t ON c.TrainerID = t.TrainerID

where c.Schedule >= '2024-06-15' and c.Schedule < '2024-06-16'

);



1. **Show the names of members who attend the classes which are instructed by the trainer whose id is=5.**

SELECT m.FirstName, m.LastName

FROM Member m

WHERE m.MemberID IN (

SELECT a.MemberID

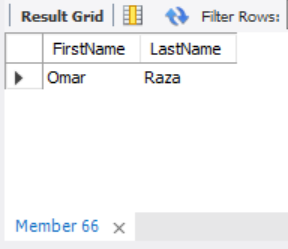
FROM Attendance AS a

Join Class AS c on a.ClassID = c.ClassID

JOIN Trainer t ON c.TrainerID = t.TrainerID

WHERE t.TrainerID = 6

);



**Joins:**

**INNER JOIN**

1. **Display information about members who attended classes along with their feedback, including details about the trainers of those classes.**

SELECT m.FirstName, m.LastName, c.ClassName, f.Rating, f.Comments, t.FirstName , t.LastName

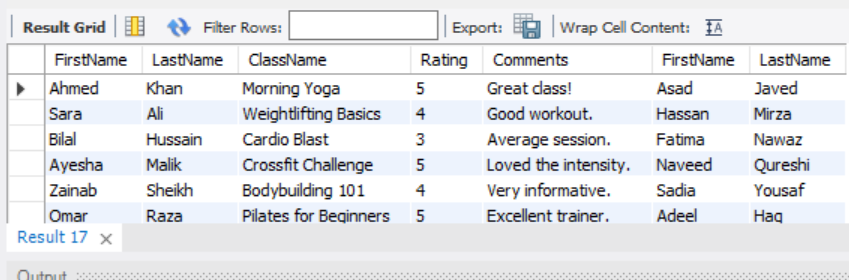
FROM Member AS m

INNER JOIN Attendance AS a ON m.MemberID = a.MemberID

INNER JOIN Class AS c ON a.ClassID = c.ClassID

INNER JOIN Feedback AS f ON m.MemberID = f.MemberID AND c.ClassID = f.ClassID

INNER JOIN Trainer AS t ON c.TrainerID = t.TrainerID;



**LEFT JOIN**

1. **Retrieve all members along with their attendance records, class details, and trainer names. Ensure that all members are included in the results even if they haven't attended any classes.**

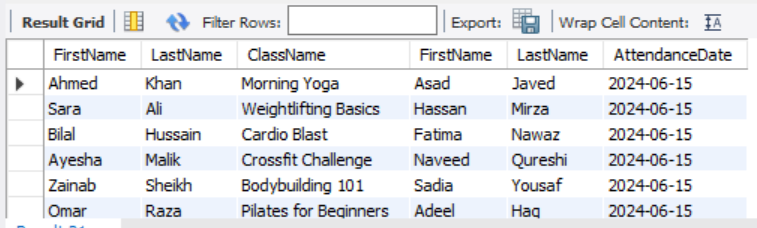
SELECT m.FirstName, m.LastName, c.ClassName, t.FirstName, t.LastName, a.AttendanceDate

FROM Member AS m

LEFT JOIN Attendance AS a ON m.MemberID = a.MemberID

LEFT JOIN Class AS c ON a.ClassID = c.ClassID

LEFT JOIN Trainer t ON c.TrainerID = t.TrainerID;



**RIGHT JOIN**

1. **Display information about classes, their schedules, trainers, members who attended those classes, and their feedback. Include all classes and attendees, even if there is no feedback provided.**

SELECT c.ClassName, c.Schedule, t.FirstName, t.LastName , m.FirstName, m.LastName,f.Rating,f.Comments

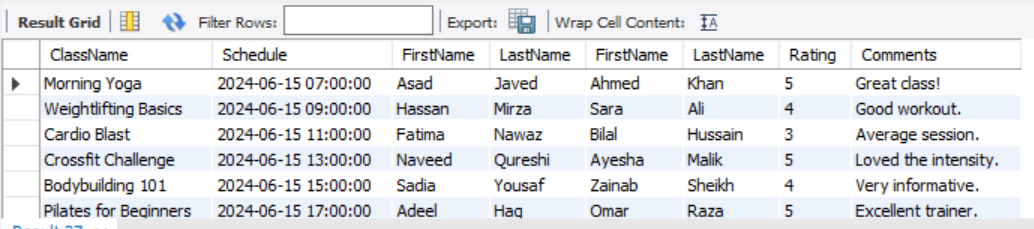
FROM Attendance AS a

RIGHT JOIN Class AS c ON a.ClassID = c.ClassID

RIGHT JOIN Member AS m ON a.MemberID = m.MemberID

RIGHT JOIN Trainer t ON c.TrainerID = t.TrainerID

LEFT JOIN Feedback f ON c.ClassID = f.ClassID AND m.MemberID = f.MemberID;



**FULL JOIN**

1. **Display information of members along with their membership as well as their trainer and classes. Show all members if they have no membership. Show all trainers if they have no class assigned.**

SELECT m.memberID, m.FirstName, NULL AS TrainerID, NULL AS TrainerFirstName, p.MembershipId, p.MembershipType, NULL AS ClassID, NULL AS ClassName

FROM MEMBER AS m

LEFT JOIN Membership AS p ON m.MembershipID=p.MembershipID

**UNION**

SELECT NULL AS memberID, NULL AS FirstName, t.TrainerId, t.FirstName, NULL AS MembershipId, NULL AS MembershipType, c.ClassId, c.Classname

FROM Trainer AS t

RIGHT JOIN Class c on t.TrainerId= c.trainerId;

