**Joins and Sub Queries:**

Great Talha! Let’s go through **Joins** and **Subqueries** in your **ride-sharing app**, interview-style — clean, simple, and real examples 🚗💬

**🔗 JOINS (combine data from 2+ tables)**

**🎯 Tables**

**Passengers**

passenger\_id | name

-------------|-------

P01 | Talha

P02 | Ali

**Rides**

ride\_id | passenger\_id | driver\_id | fare

--------|--------------|-----------|-----

1 | P01 | D01 | 300

2 | P02 | D02 | 250

**Drivers**

driver\_id | name

----------|-----

D01 | Ahmed

D02 | Bilal

**1. INNER JOIN**

Only matching rows

SELECT p.name, d.name AS driver, r.fare

FROM Rides r

INNER JOIN Passengers p ON r.passenger\_id = p.passenger\_id

INNER JOIN Drivers d ON r.driver\_id = d.driver\_id;

🎯 Output: Talha got ride with Ahmed (only matched rows)

**2. LEFT JOIN**

All passengers + their rides (even if no ride)

SELECT p.name, r.ride\_id

FROM Passengers p

LEFT JOIN Rides r ON p.passenger\_id = r.passenger\_id;

🧾 Shows all passengers, including those who haven’t booked

**3. RIGHT JOIN**

All rides + passenger names (even if passenger info missing)

SELECT r.ride\_id, p.name

FROM Rides r

RIGHT JOIN Passengers p ON p.passenger\_id = r.passenger\_id;

**4. FULL OUTER JOIN**

All from both tables; nulls where no match

SELECT p.name, r.ride\_id

FROM Passengers p

FULL OUTER JOIN Rides r ON p.passenger\_id = r.passenger\_id;

✅ Useful in logs/audits

**5. CROSS JOIN**

Every possible combo (rarely used)

SELECT p.name, d.name

FROM Passengers p

CROSS JOIN Drivers d;

👎 Not common unless building cartesian test cases

**🔍 SUBQUERIES**

**1. In SELECT clause**

SELECT name,

(SELECT COUNT(\*) FROM Rides r WHERE r.passenger\_id = p.passenger\_id) AS total\_rides

FROM Passengers p;

🚕 Shows how many rides Talha took

**2. In WHERE clause**

SELECT name

FROM Passengers

WHERE passenger\_id IN (SELECT passenger\_id FROM Rides WHERE fare > 250);

🎯 Passengers with premium fares

**3. In FROM clause (as derived table)**

SELECT p.name, ride\_count

FROM Passengers p

JOIN (

SELECT passenger\_id, COUNT(\*) AS ride\_count

FROM Rides

GROUP BY passenger\_id

) AS ride\_summary ON p.passenger\_id = ride\_summary.passenger\_id;

🙌 Shows each passenger and their total rides using a subquery