INDIAN DATA CLUB

INDIAN DATA CLUB

Day 17 (22/11): Subqueries (SELECT and FROM clause)

🎯 ***Objective***

To understand and apply **subqueries inside the SELECT and FROM clauses** to perform advanced data retrieval, derive new metrics, and build temporary result sets for complex analysis.

🔍 ***Topics Covered***

- Subqueries in the SELECT clause
- Subqueries in the FROM clause
- Scalar subqueries
- Inline views / derived tables
- Row-wise comparisons using subqueries
- Aggregations inside subqueries

INDIAN DATA CLUB

Subqueries in the SELECT Clause

Used to compare each row with an aggregated value or fetch a single value.

```
SELECT
    col1,
    (SELECT AVG(salary) FROM employees) AS avg_salary
FROM employees;
```

INDIAN DATA CLUB

Subqueries in the FROM Clause (Derived Table)

Create a temporary table inside FROM for further filtering.

```
SELECT *
FROM (
    SELECT department, COUNT(*) AS emp_count
    FROM employees
    GROUP BY department
) AS dept_summary
WHERE emp_count > 10;
```

INDIAN DATA CLUB

Scalar Subqueries

Return **only one value** (single row & single column).

```
SELECT
    name,
    (SELECT MAX(salary) FROM employees) AS highest_salary
FROM employees;
```

INDIAN DATA CLUB

Row-wise Comparison Using Subqueries

Compare each row to aggregated values.

```
SELECT name, salary
FROM employees
WHERE salary > (SELECT AVG(salary) FROM employees);
```

INDIAN DATA CLUB

Aggregations Inside Subqueries

Perform group-based calculations inside subqueries.

```
SELECT *
FROM (
    SELECT category, COUNT(*) AS total_products
    FROM products
    GROUP BY category
) AS summary;
```

INDIAN DATA CLUB

Practice Questions:

```sql
-- Show each patient with their service's average satisfaction

SELECT
    patient_id,
    name AS patient_name,
    service,
    (
        SELECT AVG(satisfaction)
        FROM patients p2
        WHERE p2.service = p1.service
    ) AS avg_satisfaction
FROM patients p1;
```

patient_id	patient_name	service	avg_satisfaction
PAT-003ce690	Larry Dixon	ICU	79.9212
PAT-00883d3c	Victor Taylor	ICU	79.9212
PAT-00b64d32	Amber Obrien	general_medicine	78.5744
PAT-029113eb	Victoria Larson	surgery	80.3150
PAT-02ae68da	Corey Whitaker	ICU	79.9212
PAT-02f2e75d	Thomas Atkins	emergency	79.5475
PAT-0307a640	Steven Howard	general_medicine	78.5744
PAT-030f2b7d	Chris Velazquez	ICU	79.9212
PAT-032160f4	Angela Lin	general_medicine	78.5744
PAT-03a1bd78	Melissa Mcfarland	emergency	79.5475
PAT-03ae75aa	Jordan Williams	surgery	80.3150
PAT-03d0b953	Michelle Davis D...	emergency	79.5475
PAT-03f5abbe	Dean Donovan	surgery	80.3150
PAT-044a304c	George Harper	surgery	80.3150
PAT-0477a000	Michael Wang	emergency	79.5475
PAT-04866ac6	Scott Alexander	surgery	80.3150
PAT-04a8031e	Jake Shaw	ICU	79.9212
PAT-04be60be	Bradley Johnso...	general_medicine	78.5744
PAT-06a8ea25	Alexander Gomez	emergency	79.5475
PAT-06d975bc	Donald Medina	general_medicine	78.5744
PAT-07a9ee88	Michael Morrison	surgery	80.3150

INDIAN DATA CLUB

Practice Questions:

```sql
-- Create a derived table of service statistics and query from it.
SELECT
    p.patient_id,
    p.name AS patient_name,
    p.service,
    s.avg_satisfaction
FROM patients p
JOIN (
        SELECT
            service,
            AVG(satisfaction) AS avg_satisfaction
        FROM patients
        GROUP BY service
     ) AS s
ON p.service = s.service;
```

patient_id	patient_name	service	avg_satisfaction
PAT-003ce690	Larry Dixon	ICU	79.9212
PAT-00883d3c	Victor Taylor	ICU	79.9212
PAT-00b64d32	Amber Obrien	general_medicine	78.5744
PAT-029113eb	Victoria Larson	surgery	80.3150
PAT-02ae68da	Corey Whitaker	ICU	79.9212
PAT-02f2e75d	Thomas Atkins	emergency	79.5475
PAT-0307a640	Steven Howard	general_medicine	78.5744
PAT-030f2b7d	Chris Velazquez	ICU	79.9212
PAT-032160f4	Angela Lin	general_medicine	78.5744
PAT-03a1bd78	Melissa Mcfarland	emergency	79.5475
PAT-03ae75aa	Jordan Williams	surgery	80.3150
PAT-03d0b953	Michelle Davis D...	emergency	79.5475
PAT-03f5abbe	Dean Donovan	surgery	80.3150
PAT-044a304c	George Harper	surgery	80.3150
PAT-0477a000	Michael Wang	emergency	79.5475
PAT-04866ac6	Scott Alexander	surgery	80.3150
PAT-04a8031e	Jake Shaw	ICU	79.9212
PAT-04be60be	Bradley Johnso...	general_medicine	78.5744
PAT-06a8ea25	Alexander Gomez	emergency	79.5475
PAT-06d975bc	Donald Medina	general_medicine	78.5744
PAT-07a9ee88	Michael Morrison	surgery	80.3150

INDIAN DATA CLUB

Practice Questions:

```sql
-- Display staff with their service's total patient count as a calculated field.
SELECT
    s.staff_id,
    s.staff_name,
    s.service,
    p.total_patient_count
FROM staff s
JOIN (
        SELECT
            service,
            COUNT(patient_id) AS total_patient_count
        FROM patients
        GROUP BY service
     ) AS p
ON s.service = p.service;
```

staff_id	staff_name	service	total_patient_count
STF-00b6381d	Steve Sanchez	ICU	241
STF-00fbd582	John Pierce	surgery	254
STF-0196d344	Denise Jacobs	ICU	241
STF-021fdf9f	Sharon Cochran	ICU	241
STF-02ae59ca	Noah Rhodes	emergency	263
STF-052894a3	Richard Rodriguez	ICU	241
STF-05591498	William Herrera	ICU	241
STF-094f410b	Ryan Munoz	emergency	263
STF-0aaed714	Rebecca Henderson	surgery	254
STF-0b0dfb38	Mike Allen	ICU	241
STF-107a58e4	Cristian Santos	emergency	263
STF-130577e6	Dylan Miller	emergency	263
STF-13f243f8	Paula Moreno	surgery	254
STF-15269c02	Sherry Decker	surgery	254
STF-15c07995	Anthony Rodriguez	surgery	254
STF-177e5df8	Angelica Tucker	surgery	254
STF-1881ede3	Aaron Wise	ICU	241
STF-1ad309f8	Matthew Foster	emergency	263
STF-1b299895	Christina Walters	ICU	241
STF-1bdb5ff4	Deborah Figueroa	ICU	241
STF-1ded4330	Joshua Blair	surgery	254

INDIAN DATA CLUB

Daily Challenge:

```sql
-- Report for each service with total admissions,
-- difference from overall average, and ranking indicator

SELECT
    service,
    total_admissions,
    (total_admissions - overall_avg_admissions) AS diff_from_avg,
    CASE
        WHEN total_admissions > overall_avg_admissions THEN 'Above Average'
        WHEN total_admissions = overall_avg_admissions THEN 'Average'
        ELSE 'Below Average'
    END AS rank_indicator
FROM (
        SELECT
            service,
            SUM(patients_admitted) AS total_admissions
        FROM services_weekly
        GROUP BY service
    ) AS service_totals
CROSS JOIN (
        SELECT AVG(total_admissions) AS overall_avg_admissions
        FROM (
                SELECT
                    service,
                    SUM(patients_admitted) AS total_admissions
                FROM services_weekly
                GROUP BY service
            ) AS t
    ) AS avg_table
ORDER BY total_admissions DESC;
```

service	total_admissions	diff_from_avg	rank_indicator
general_medicine	2332	869.2500	Above Average
surgery	1686	223.2500	Above Average
emergency	1185	-277.7500	Below Average
ICU	648	-814.7500	Below Average