🧠 Day 68 – SQL 100 Days Challenge Reflection

Dataset Theme: *Hospital Management System – Patients, Doctors, Appointments & Bills*

✅ Learning Overview

Today's dataset simulated a real-world healthcare setup. It tested my ability to combine **aggregation, joins, window functions, and nested CTEs** effectively — particularly focusing on hospital KPIs and city-wise analytics.

🧩 Key Learnings

1. **JOIN + Aggregation:** Reinforced how inner joins between multiple tables can provide meaningful hospital-level insights.

2. **CTE & RANK():** Ranking doctors based on revenue earned helped me solidify window function logic.

3. **CASE Statements:** Classifying doctors by revenue bracket improved understanding of conditional grouping.

4. **Correlated Subquery:** Identifying doctors above hospital-average revenue deepened my analytical approach.

5. **LAG() + DATEDIFF():** Realized the power of date functions in tracking patient revisit intervals.

6. **Nested CTEs:** Finding the top-performing doctor per city felt like a true analytical challenge.

7. **Advanced KPI Query:** Calculating *repeat patient percentage* for each doctor was both practical and rewarding.

⚙️ Challenges Faced

- **Bonus Query** (Most Profitable City) was the toughest part — required combining ranking logic and city-level revenue aggregation creatively.

- Keeping query structure readable and optimized took effort due to multi-step joins and ranking layers.

💡 Reflection

Comfort with **window functions (RANK, LAG)** and **analytical CTE patterns** has significantly improved. I can now confidently structure complex queries across domains — finance, retail, and healthcare.

🔁 Next Focus

Continue refining optimization and readability of multi-level SQL logic before moving into advanced performance tuning concepts.