SQL DEVELOPER - TASK 3

Task 3: Subqueries and Aggregations

Objective: Use subqueries to extract insights from a dataset and perform data aggregations to summarize and analyze the data.

Step 1: Database Setup

We create a **Students** table with the following fields:
• student_id (Primary Key)
• name
• math_score
• science_score
• english_score
• total_score (calculated dynamically)

Step 2: Sub Tasks

Task 1: Identify Top Students by Total Scores

SQL Query:
SELECT name, (math_score + science_score + english_score) AS total_score FROM Students ORDER BY total_score DESC LIMIT 5;
Output:

Student	TotalScore
Ananya Desai	275
Rohan Verma	273
Aditya Rao	268
Aarav Sharma	265
Meera Joshi	263

Task 2: Calculate Averages Based on Conditions

Example Queries:
1. SELECT AVG(math_score) FROM Students WHERE math_score > 70;
2. SELECT AVG(math_score + science_score + english_score) FROM Students WHERE (math_score + science_score + english_score) BETWEEN 200 AND 250;
Outputs:

Metric	Value
AvgMathAbove70	85.375
AvgTotal_200_250	205.0

ScoreRange	StudentCount	AvgTotalInRange
150–199	1	187.0
200–249	2	205.0
250+	7	265.29

Task 3: Find Second-Highest Math Scores

SQL Query:
SELECT MAX(math_score) FROM Students WHERE math_score < (SELECT MAX(math_score) FROM Students);
Output:

Metric	Value
SecondHighestMath	92

StudentID	Name	MathScore
1	Aarav Sharma	92

Summary & Conclusion

In Task 3, subqueries and aggregations were used to analyze student performance. We identified top students, calculated averages under conditions, and determined the second-highest Math score. These techniques help extract insights from raw datasets.