**Name:** Manju

**Date:** 25/06/2025

**Batch:** May 25th Batch

This document includes a set of SQL queries executed as part of the assigned **Week 4 task.**

The queries demonstrate the use of window functions such as RANK(), SUM() OVER, DENSE\_RANK(), and PERCENT\_RANK() to perform analytical operations on student performance data.

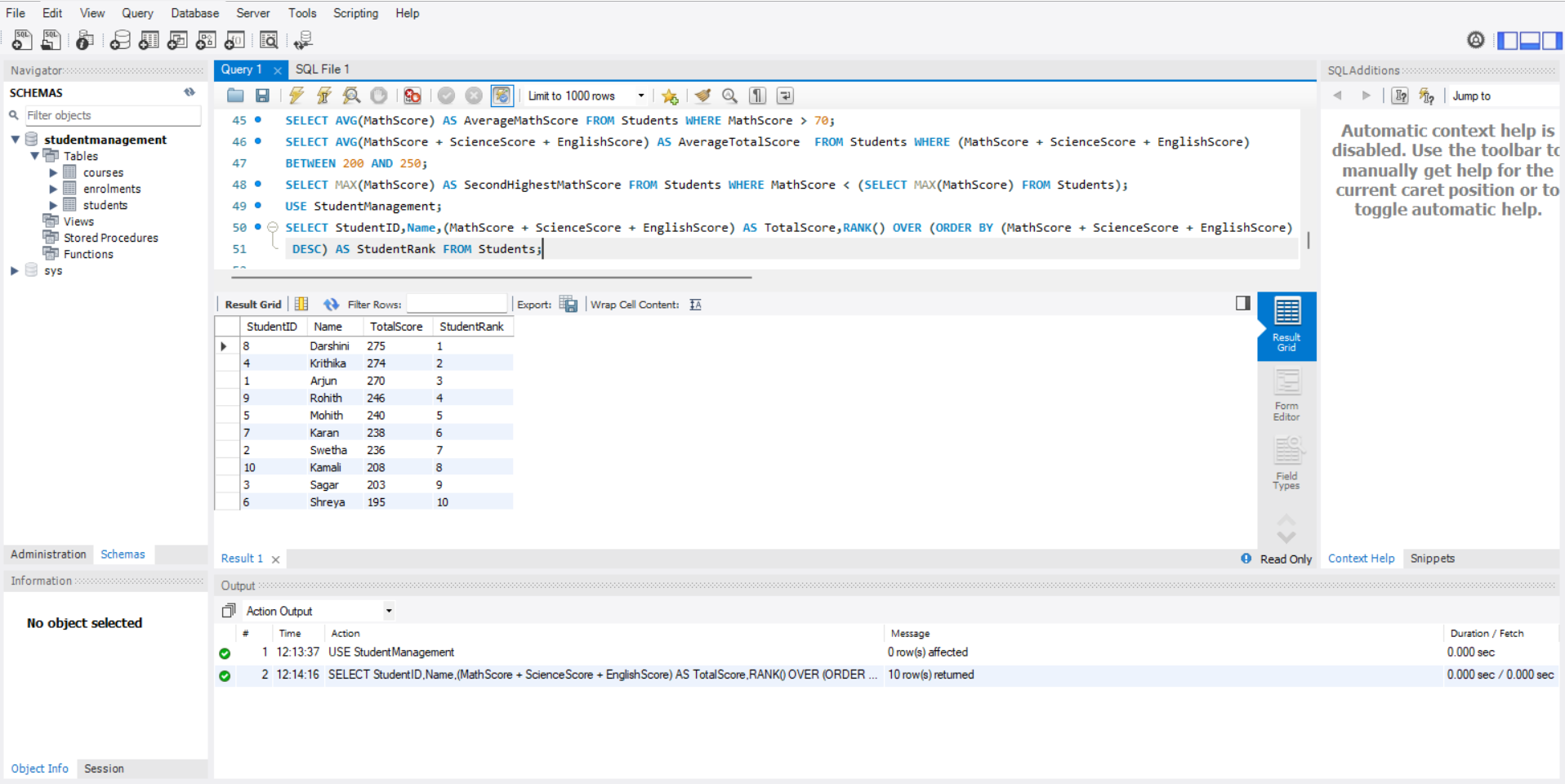
Each query is accompanied by a brief explanation and its corresponding output screenshot, as required in the assignment guidelines.

**SQL Queries**

**1.Rank Students by Total Scores:**

SELECT StudentID,Name,(MathScore + ScienceScore + EnglishScore) AS TotalScore,RANK() OVER (ORDER BY (MathScore + ScienceScore + EnglishScore) DESC) AS StudentRank FROM Students;

**Output:**



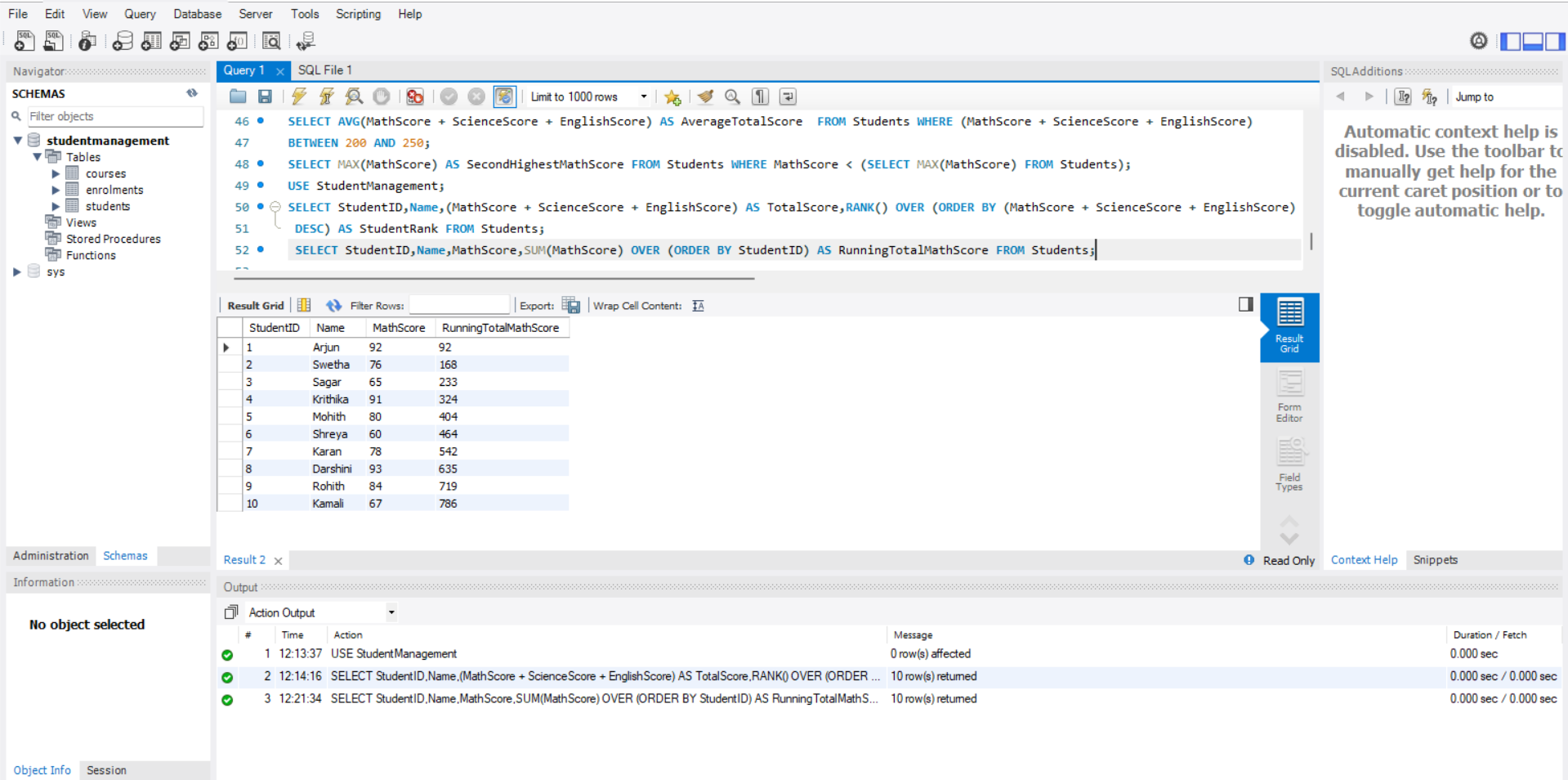
**Purpose:** To assign a rank to each student based on their total score across all subjects.

**Explanation:** Uses RANK() OVER (ORDER BY TotalScore DESC) to rank students from highest to lowest based on combined scores of Math, Science, and English.

**2.Calculate Running Total of Math Scores**

SELECT StudentID,Name,MathScore,SUM(MathScore) OVER (ORDER BY StudentID) AS RunningTotalMathScore FROM Students;

**Output:**



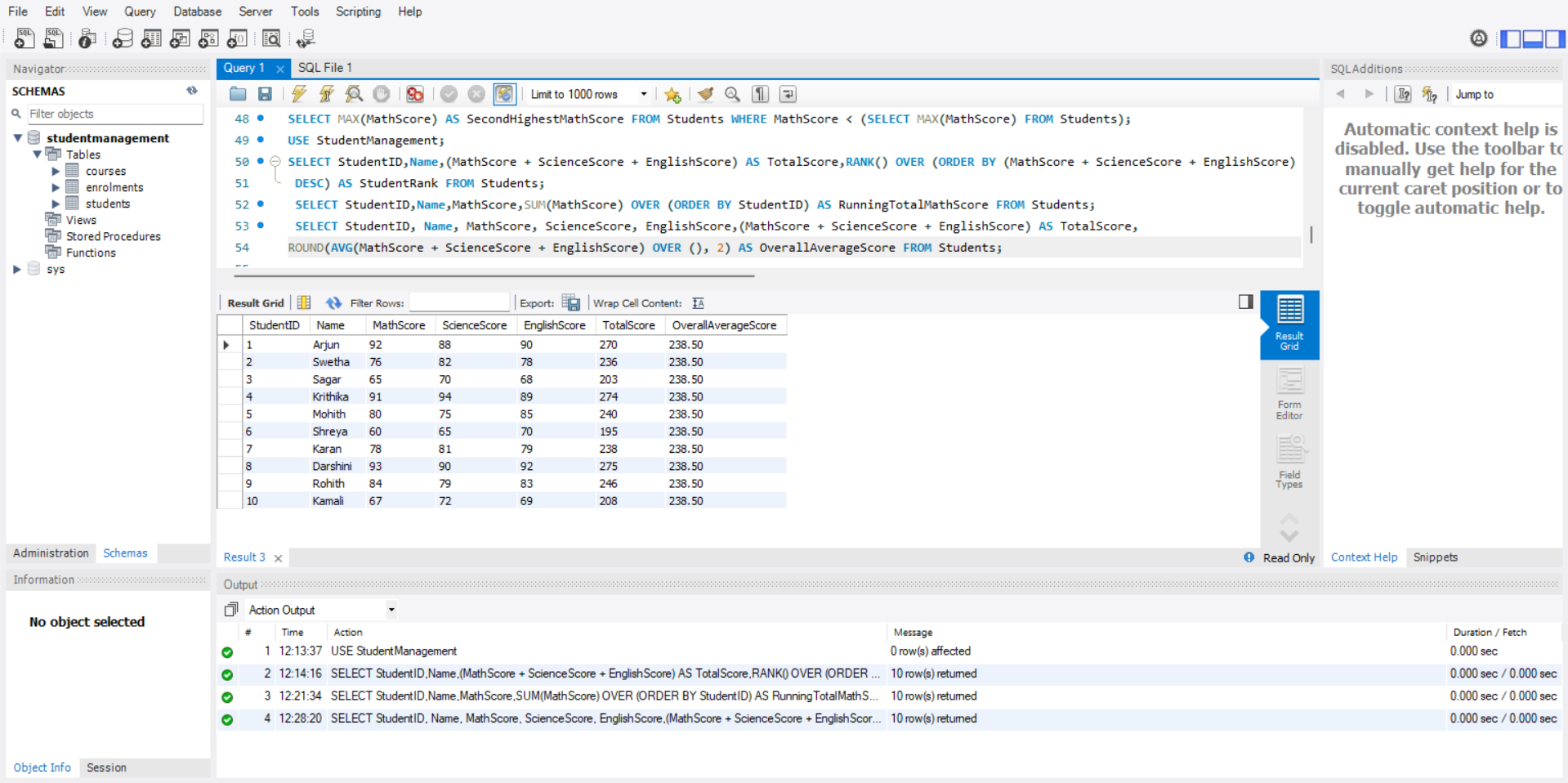
**Purpose:** To calculate the cumulative Math scores in the order of StudentID.

**Explanation:** SUM(MathScore) OVER (ORDER BY StudentID) adds Math scores progressively for each student based on StudentID sequence.

**3.Overall Average of Total Scores Using Window Function**

SELECT StudentID, Name, MathScore, ScienceScore, EnglishScore,(MathScore + ScienceScore + EnglishScore) AS TotalScore,ROUND(AVG(MathScore + ScienceScore + EnglishScore) OVER (), 2) AS OverallAverageScore FROM Students;

**Output:**

****

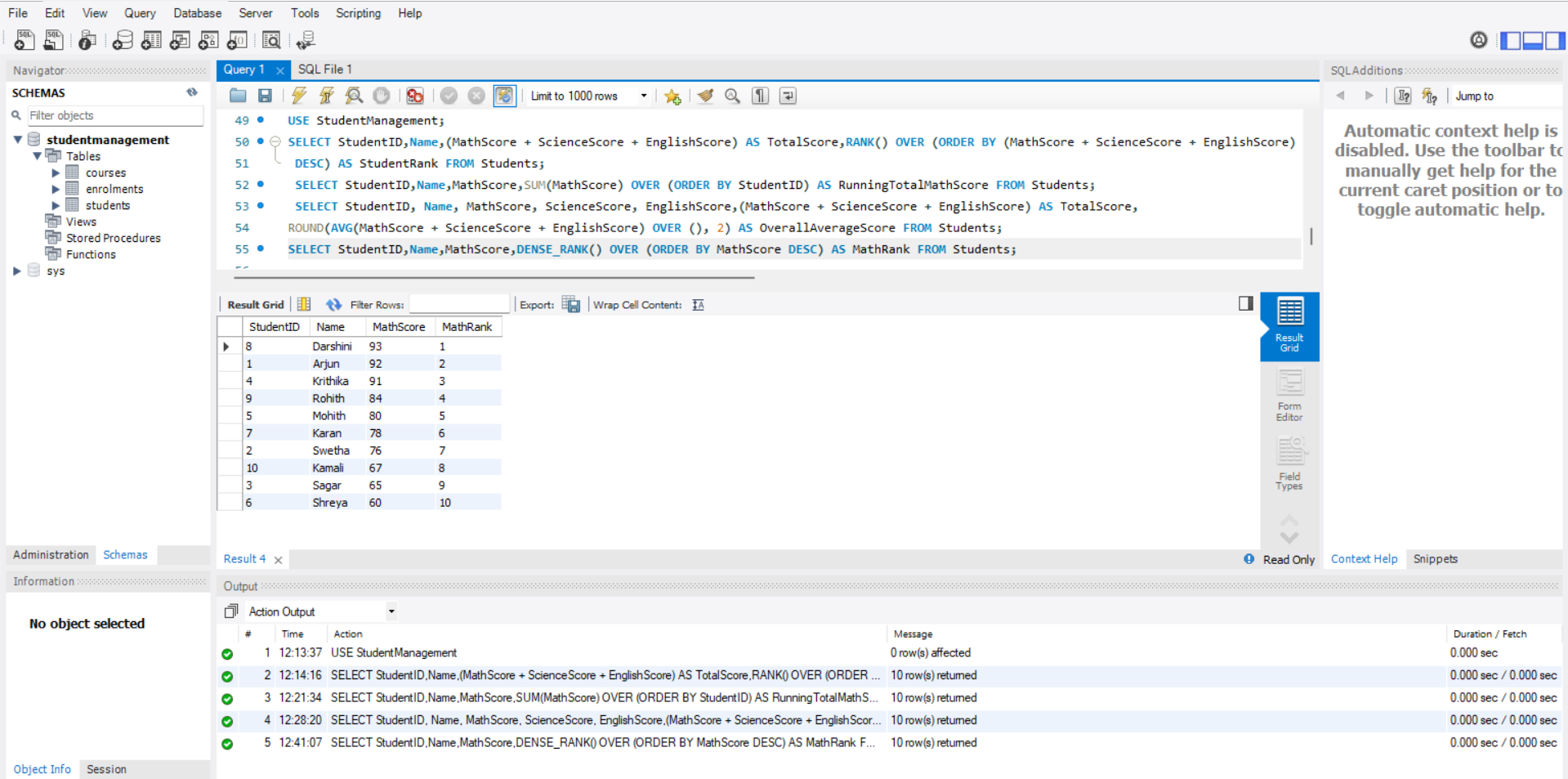
**Purpose**: To display each student’s total score along with the average total score of all students.

**Explanation**: Uses AVG(...) OVER () to compute a global average, repeated across all rows for comparison.

**4. Rank Students by Math Score Using DENSE\_RANK**

SELECT StudentID,Name,MathScore,DENSE\_RANK() OVER (ORDER BY MathScore DESC) AS MathRank FROM Students;

**Output:**

****

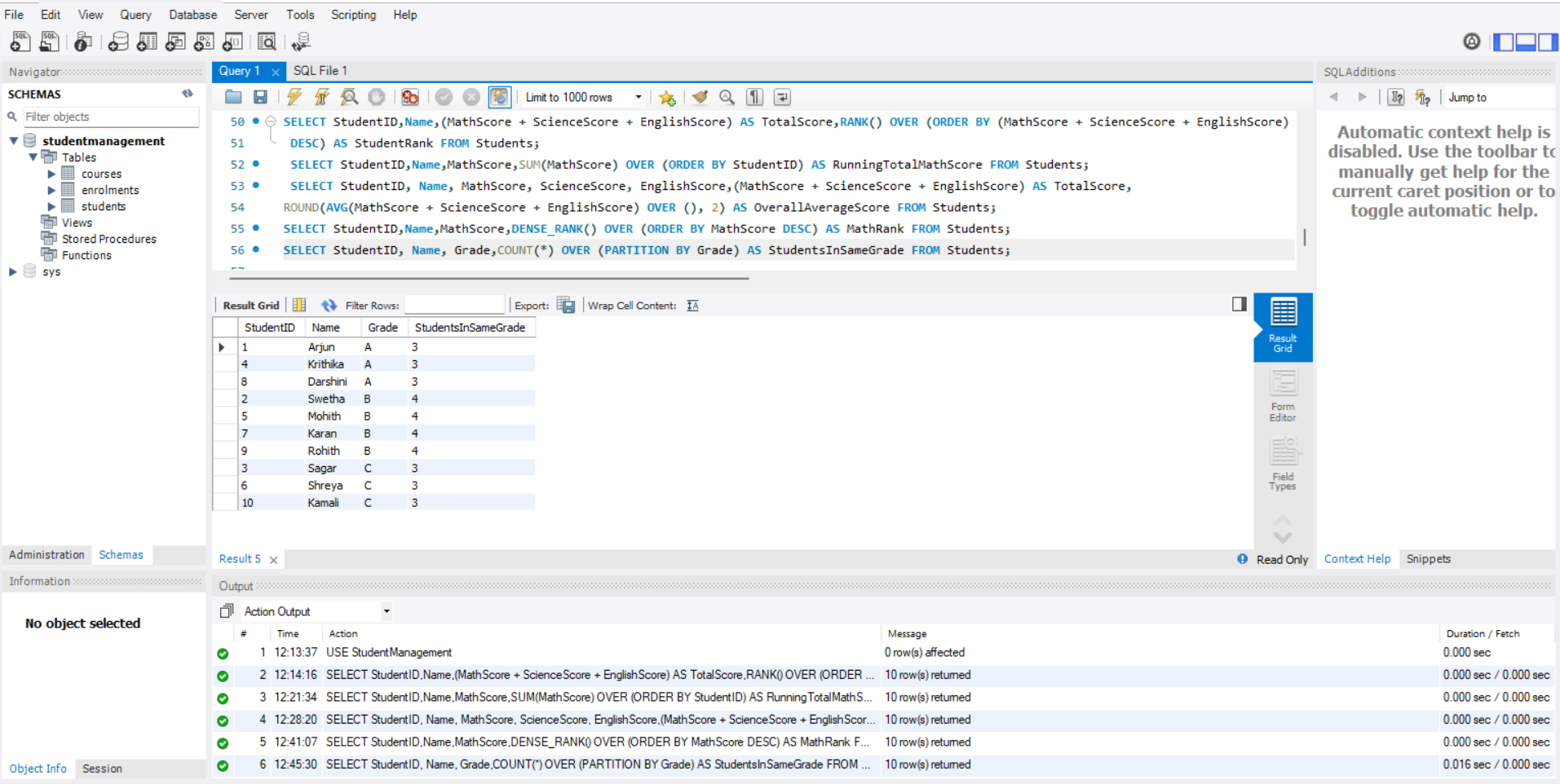
**Purpose**: Rank Students by Math Score Using DENSE\_RANK

**Explanation**: DENSE\_RANK() gives the same rank for equal scores and does not skip ranks, useful when handling ties.

**5. Count of Students in Each Grade**

SELECT StudentID, Name, Grade,COUNT(\*) OVER (PARTITION BY Grade) AS StudentsInSameGrade FROM Students;

**Output:**



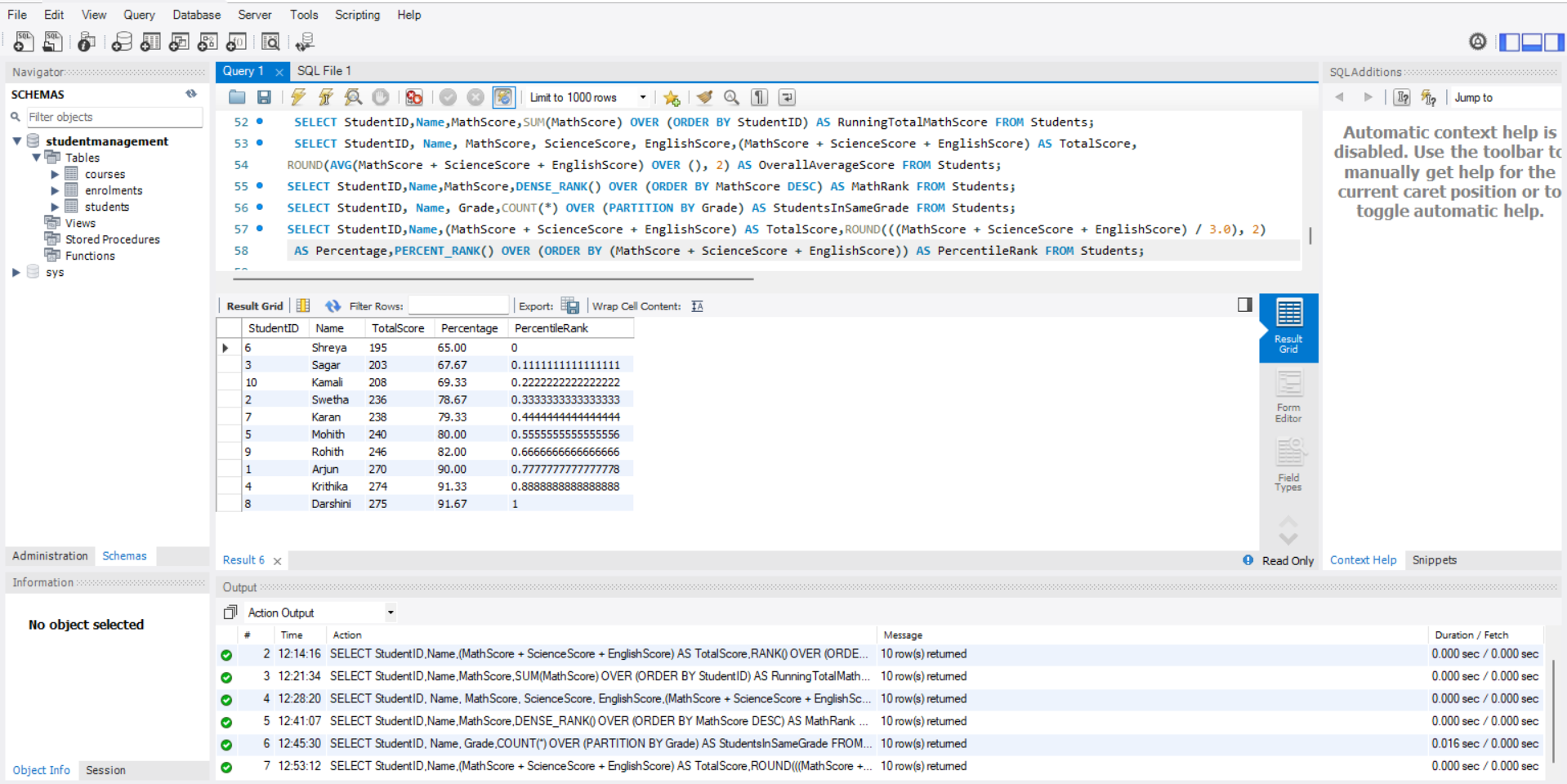
**Purpose**: To find out how many students are in each grade category.

**Explanation:** COUNT(\*) OVER (PARTITION BY Grade) groups students by Grade and counts how many are in each group.

**6. Calculate Percentage and Percentile Rank Based on Total Score**

SELECT StudentID,Name,(MathScore + ScienceScore + EnglishScore) AS TotalScore,ROUND(((MathScore + ScienceScore + EnglishScore) / 3.0), 2) AS Percentage,PERCENT\_RANK() OVER (ORDER BY (MathScore + ScienceScore + EnglishScore)) AS PercentileRank FROM Students;

**Output:**



**Purpose:** To calculate each student’s percentage and their relative standing (percentile) in the class.

**Explanation:** Combines average score per student with PERCENT\_RANK() to show how they rank in percentage terms compared to others.

**Summary of Findings**

**Top Students Identified:**We ranked the students based on their total scores, which helped us spot the top performers in the class. Even when some students had the same score, the RANK() function made it easy to see who stood out overall.

**Tracking Math Progress:**By using the SUM() function to get a running total of Math scores, we could see how each student contributed to the subject's overall performance. It gave us a clear picture of progress in Math as we moved down the list.

**Class Average Comparison:**With the help of the AVG() function, we calculated the class average for total scores. This made it simple to compare each student’s performance against the average, showing who was above or below the class standard.

**Math-Specific Rankings:**We used DENSE\_RANK() on the Math scores to highlight students who are especially strong in Math, regardless of how they did in other subjects. This helped us find subject-specific toppers.

**Distribution by Grade:**Using COUNT(\*) OVER (PARTITION BY Grade), we figured out how many students are in each grade. This gave us an idea of how students are grouped and how the academic workload might vary across grades.

**Percentage and Percentile Insights:**By calculating each student's percentage and percentile rank, we got a clearer sense of how they compare to their classmates. This helped us evaluate performance in a fair and meaningful way.