

COMPLEX QUERY

1. High-Attendance Students (Using a CTE and the existing CalculateAttendance function)

English Question: Which students have an overall attendance percentage of 75% or higher across *all* their enrolled courses?

Calculates the student's average attendance percentage across all their enrolled courses by dynamically calculating the percentage for each course first (using the Derived Table), and then averaging the results.

2. Department Course Load View (Using a View)

English Question: Create a permanent view to show each department's total number of courses, the sum of all course credits offered, and the department's chairperson.

Provides an immediate, readable summary of the department's total teaching load (course count and total credits offered) and who the chairperson is.

3. Course-Wise Grade Distribution (Using a CTE and Conditional Aggregation)

English Question: For a specific course and assessment, what is the count of students who received an 'A' grade or higher ('S' or 'A') versus those who received a passing grade ('B' through 'E')?

Calculates the exact counts of students falling into high-grade, passing, and failing categories for a single assessment in a single course.

4. Teachers without a Course (Using LEFT JOIN and WHERE IS NULL)

English Question: Which teachers are currently not assigned to teach any course?

Identifies all teachers who do not have an active teaching assignment (i.e., their ID does not appear in the list of instructors in the Courses table).

5. Student Performance and Course Load Ranking (Using Window Functions)

English Question: List every student, their total course credits, and their average marks for a specific assessment, then rank them based on their average marks (highest is rank 1).

Ranks every student based on their average marks for a specific assessment type, while also showing their total course load

TRIGGERS

- Role_update: Logs a record into the User_Audit_Log table whenever a user's role is changed (updated) in the Users table.
- trg_before_department_delete: Prevents the deletion of a department if there are any active courses currently assigned to it in the Courses table. It raises a custom SQL error if a course exists.
- Trg_before_grade_insert_update: Automatically calculates and sets the letter grade ('S' through 'F') for a new grade record *before* it is inserted, based on the marks_obtained as a percentage of total_marks

FUNCTIONS

- CalculateAttendance: Calculates and returns the **attendance percentage** for a specific student in a specific course: (Present Days / Total Days) * 100. Returns 0.00 if no attendance records exist.

- GetTotalCourseCredits :Calculates and returns the **sum of credits** for all courses a specified student is currently enrolled in. Returns 0 if the student is not enrolled in any course.

PROCEDURE

- Student_Courses: Retrieves a list of **all courses** a given student is enrolled in, including the course ID, name, credits, and the name of the assigned teacher.
- GetCoursePerformanceSummary: Provides a summary of performance for a specific course and assessment, showing the **total enrolled students**, the **average marks obtained**, and the **maximum total marks** for that assessment.
- GetStudentsHighOverallAttendance: Lists all students whose **average attendance percentage across all their enrolled courses** meets or exceeds the specified minimum percentage.
- GetCourseGradeDistributionStats: Calculates and shows the distribution of grades for a specific course and assessment, broken down into counts for **High Grades ('S', 'A')**, **Passing Grades ('B' to 'E')**, and **Failing Grades ('F')**.
- GetUnassignedTeachers: Lists all teachers who are **not currently assigned to teach any course** in the Courses table
- GetStudentAssessmentRanking: Lists all students, their total enrolled credits, their **average marks** for the specified assessment type, and provides a **performance rank** based on those average marks (highest marks get rank 1).