Task 4: Aggregate Functions and Grouping

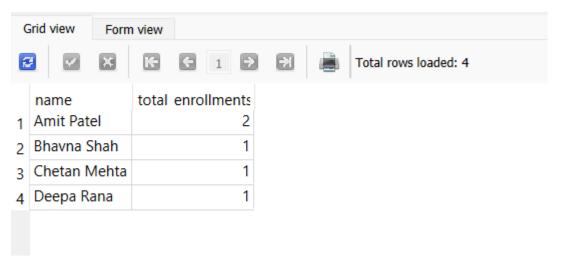
COUNT() – Count number of enrollments per student

SELECT s.name, COUNT(e.enrollment_id) AS total_enrollments

FROM enrollments e

JOIN students s ON e.student_id = s.student_id

GROUP BY s.name;



SUM() - Total credits of courses each student has enrolled in

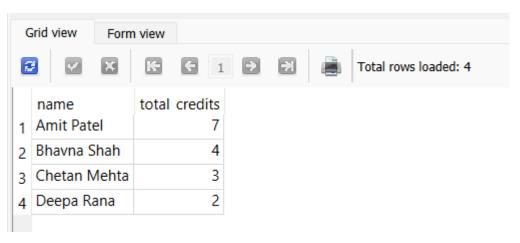
SELECT s.name, SUM(c.credits) AS total_credits

FROM enrollments e

JOIN students s ON e.student_id = s.student_id

JOIN courses c ON e.course_id = c.course_id

GROUP BY s.name;



.....

AVG() - Average credits of courses each student has enrolled in

SELECT s.name, AVG(c.credits) AS avg_credits

FROM enrollments e

JOIN students s ON e.student_id = s.student_id

JOIN courses c ON e.course_id = c.course_id

GROUP BY s.name;



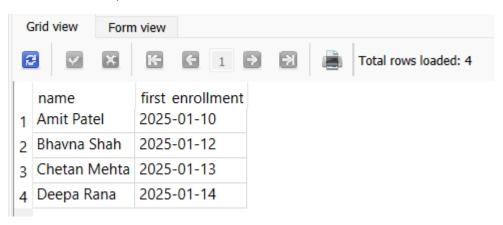
MIN() – Find the earliest enrollment date for each student

SELECT s.name, MIN(e.enrollment_date) AS first_enrollment

FROM enrollments e

JOIN students s ON e.student_id = s.student_id

GROUP BY s.name;



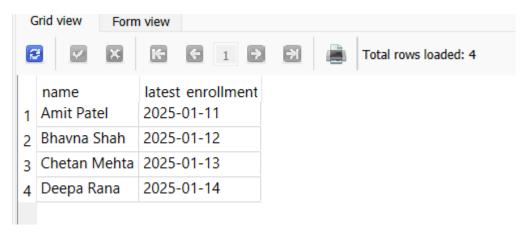
MAX() - Find the latest enrollment date for each student

SELECT s.name, MAX(e.enrollment_date) AS latest_enrollment

FROM enrollments e

JOIN students s ON e.student_id = s.student_id

GROUP BY s.name;



Multiple Aggregates Together

SELECT s.name,

COUNT(e.enrollment_id) AS total_courses,

SUM(c.credits) AS total_credits,

AVG(c.credits) AS avg_credits,

MIN(e.enrollment_date) AS first_enrollment,

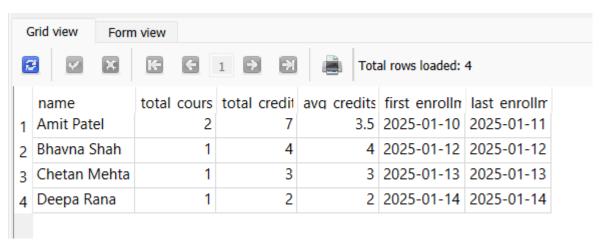
MAX(e.enrollment_date) AS last_enrollment

FROM enrollments e

JOIN students s ON e.student_id = s.student_id

JOIN courses c ON e.course_id = c.course_id

GROUP BY s.name;



SELECT s.name, COUNT(e.enrollment_id) AS total_enrollments

FROM enrollments e

JOIN students s ON e.student_id = s.student_id

GROUP BY s.name

HAVING COUNT(e.enrollment_id) > 1;

