**Student Information System (SIS)**

**Task 4. Subquery and its type:**

**1. Write an SQL query to calculate the average number of students enrolled in each course. Use aggregate functions and subqueries to achieve this.**

mysql> select avg(student\_count) as average\_enrollment

-> from (

-> select course\_id, count(student\_id) as student\_count

-> from enrollments

-> group by course\_id

-> ) as course\_enrollment\_counts;

**2. Identify the student(s) who made the highest payment. Use a subquery to find the maximum payment amount and then retrieve the student(s) associated with that amount.**

mysql> select s.student\_id, s.first\_name, s.last\_name, p.amount

-> from students s

-> join payments p on s.student\_id = p.student\_id

-> where p.amount = (select max(amount) from payments);

**3. Retrieve a list of courses with the highest number of enrollments. Use subqueries to find the course(s) with the maximum enrollment count.**

mysql> select c.course\_id, c.course\_name, count(e.student\_id) as enrollment\_count

-> from courses c

-> join enrollments e on c.course\_id = e.course\_id

-> group by c.course\_id, c.course\_name

-> having count(e.student\_id) = (

-> select max(course\_enrollment)

-> from (select count(student\_id) as course\_enrollment from enrollments group by course\_id) as subquery

-> );

**4. Calculate the total payments made to courses taught by each teacher. Use subqueries to sum payments for each teacher's courses.**

mysql> select t.teacher\_id, t.first\_name, t.last\_name, sum(p.amount) as total\_payments

-> from teacher t

-> join courses c on t.teacher\_id = c.teacher\_id

-> join enrollments e on c.course\_id = e.course\_id

-> join payments p on e.student\_id = p.student\_id

-> group by t.teacher\_id, t.first\_name, t.last\_name;

**5. Identify students who are enrolled in all available courses. Use subqueries to compare a student's enrollments with the total number of courses.**

mysql> select s.student\_id, s.first\_name, s.last\_name

-> from students s

-> where (select count(distinct course\_id) from enrollments where student\_id = s.student\_id) =

-> (select count(course\_id) from courses);

**6. Retrieve the names of teachers who have not been assigned to any courses. Use subqueries to find teachers with no course assignments.**

mysql> select t.teacher\_id, t.first\_name, t.last\_name

-> from teacher t

-> where not exists (select 1 from courses where teacher\_id = t.teacher\_id);

**7. Calculate the average age of all students. Use subqueries to calculate the age of each student based on their date of birth.**

mysql> select avg(year(curdate()) - year(date\_of\_birth)) as average\_age

-> from students;

**8. Identify courses with no enrollments. Use subqueries to find courses without enrollment records.**

mysql> select course\_id, course\_name

-> from courses

-> where course\_id not in (select distinct course\_id from enrollments);

**9. Calculate the total payments made by each student for each course they are enrolled in. Use subqueries and aggregate functions to sum payments.**

mysql> select e.student\_id, s.first\_name, s.last\_name, e.course\_id, c.course\_name, sum(p.amount) as total\_paid

-> from enrollments e

-> join students s on e.student\_id = s.student\_id

-> join courses c on e.course\_id = c.course\_id

-> join payments p on e.student\_id = p.student\_id

-> group by e.student\_id, e.course\_id;

**10. Identify students who have made more than one payment. Use subqueries and aggregate functions to count payments per student and filter for those with counts greater than one.**

mysql> select s.student\_id, s.first\_name, s.last\_name, count(p.payment\_id) as payment\_count

-> from students s

-> join payments p on s.student\_id = p.student\_id

-> group by s.student\_id

-> having count(p.payment\_id) > 1;

**11. Write an SQL query to calculate the total payments made by each student. Join the "Students" table with the "Payments" table and use GROUP BY to calculate the sum of payments for each student.**

mysql> select s.student\_id, s.first\_name, s.last\_name, sum(p.amount) as total\_payments

-> from students s

-> join payments p on s.student\_id = p.student\_id

-> group by s.student\_id;

**12. Retrieve a list of course names along with the count of students enrolled in each course. Use JOIN operations between the "Courses" table and the "Enrollments" table and GROUP BY to count enrollments.**

mysql> select c.course\_name, count(e.student\_id) as student\_count

-> from courses c

-> left join enrollments e on c.course\_id = e.course\_id

-> group by c.course\_id, c.course\_name;

**13. Calculate the average payment amount made by students. Use JOIN operations between the "Students" table and the "Payments" table and GROUP BY to calculate the average.**

mysql> select s.student\_id, s.first\_name, s.last\_name, avg(p.amount) as average\_payment

-> from students s

-> join payments p on s.student\_id = p.student\_id

-> group by s.student\_id, s.first\_name, s.last\_name;