📘 MySQL Homework Assignment

Part A – Foundations

1. Create a database called SchoolDB.
2. Create a table Students with the fields:
    - id (INT, primary key, auto increment)
    - first_name (VARCHAR(50))
    - last_name (VARCHAR(50))
    - age (INT)
    - gpa (DECIMAL(3,2))
3. Insert at least 5 rows into Students.
4. Write a query to select only the first_name and last_name columns.
5. Write a query to return students with gpa > 3.0.
6. Write a query to sort students by age descending.
7. Write a query to count the total number of students.
8. Write a query to find the average GPA of all students.
9. Write a query to select students whose last name starts with "A".
10. Write a query to update one student's GPA.
11. Write a query to delete a student by id.

Part B – Advanced / Challenging

12. Start a transaction that updates multiple rows, then roll it back.
13. Disable autocommit mode, perform several inserts, then commit manually.
14. Write a query that performs multiple-table update (update one table based on another).
15. Create a stored function that calculates whether a student is "Pass" or "Fail" based on GPA. Use it in a SELECT.
16. Create a stored procedure that inserts a new student, but prevents insertion if age < 10.
17. Add a table Courses and a table Enrollments (many-to-many relation). Insert sample data and write a query to get all students with their enrolled courses.
18. Add a foreign key from Enrollments.student_id to Students.id with ON DELETE CASCADE. Demonstrate what happens if you delete a student.

19. **Add another foreign key with ON UPDATE CASCADE, then update a student id to see the effect.**
20. **Create a trigger that logs every deletion from Students into a new table Students_Deleted.**
21. **Create a view called HighAchievers that shows students with GPA above 3.5.**
22. **Write a query that groups students by age and counts how many are in each group.**
23. **Write a query that finds the student(s) with the highest GPA using a subquery.**