MySQL (Day2):

1	***Display the full name (first, last) of the student with his grade.*** ***if his grade is greater than 85% Excellent, from 75% to 85% Very good, from 65% to 75% Good and from 55% to 65% pass otherwise will be graded as failed.***
	SELECT CONCAT(s.first_name, ' ', s.last_name) AS full_name, CASE WHEN sc.grade > 85 THEN 'Excellent' WHEN sc.grade BETWEEN 75 AND 85 THEN 'Very good' WHEN sc.grade BETWEEN 65 AND 75 THEN 'Good' WHEN sc.grade BETWEEN 55 AND 65 THEN 'Pass' ELSE 'Failed' END AS grade_classification FROM students s JOIN students_courses sc ON s.student_id = sc.student_id;
2	***Display the capitalized last name, and the grade, if he has no grade display the keyword absent. [using ifNULL function]***
	SELECT CONCAT(s.first_name, ' ', UPPER(s.last_name)) AS full_name, IFNULL(CASE WHEN sc.grade > 85 THEN 'Excellent' WHEN sc.grade BETWEEN 75 AND 85 THEN 'Very good' WHEN sc.grade BETWEEN 65 AND 75 THEN 'Good' WHEN sc.grade BETWEEN 55 AND 65 THEN 'Pass' ELSE 'Failed' END, 'Absent') AS grade_classification FROM students s LEFT JOIN students_courses sc ON s.student_id = sc.student_id;
3	***Display students' names, course name along with their grades.***
	SELECT CONCAT(s.first_name, ' ', s.last_name) AS full_name, c.course_name, IFNULL(CASE WHEN sc.grade > 85 THEN 'Excellent' WHEN sc.grade BETWEEN 75 AND 85 THEN 'Very good' WHEN sc.grade BETWEEN 65 AND 75 THEN 'Good' WHEN sc.grade BETWEEN 55 AND 65 THEN 'Pass' ELSE 'Failed' END, 'Absent') AS grade_classification FROM students s CROSS JOIN courses c LEFT JOIN students_courses sc ON s.student_id = sc.student_id AND c.course_id = sc.course_id;
4	***For each course, display the course name, min grade, max grade, average grade, number of attended students.***
	SELECT c.course_name, MIN(sc.grade) AS min_grade, MAX(sc.grade) AS max_grade, AVG(sc.grade) AS average_grade, COUNT(sc.student_id) AS number_of_attended_students FROM courses c LEFT JOIN students_courses sc ON c.course_id = sc.course_id GROUP BY c.course_name;

5	***Use subquery to display the <u>names of the students</u> who were born before student no 1.***
	SELECT CONCAT(first_name, ' ', last_name) AS student_name students WHERE birth_date < (SELECT birth_date FROM students WHERE student_id = 1);
6	***Use subquery to display the <u>data of all the courses</u> with a credit hour similar to MySQL's credit hours***
	SELECT course_id, course_name, credit_hour FROM courses WHERE credit_hour = (SELECT credit_hour FROM courses WHERE course_name = 'MySQL');