# **Tutorial 6 -SQL Exercises 4**

**Task 3**

Based on the relational database tables created in above, write *SQL* queries and run them for the following questions/cases (questions 1-14). Give meaningful names to columns in the output/result table of each query.

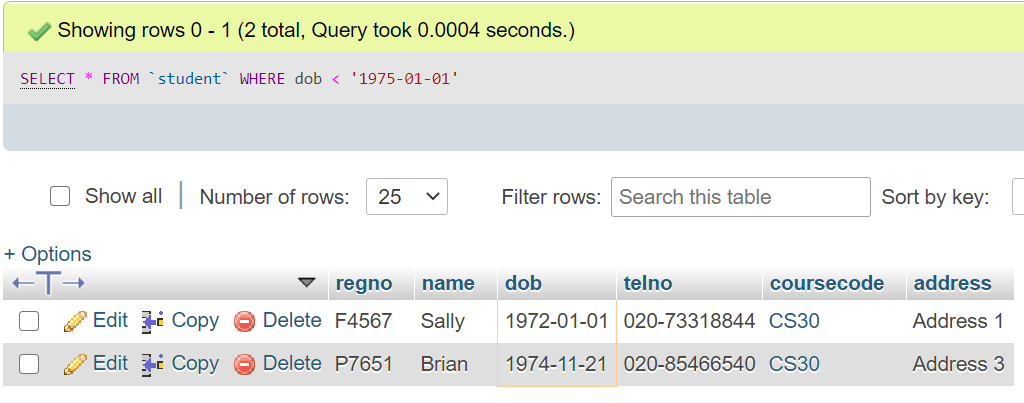
**Question 1:**

List all students whose date of birth is before 1975.

**SELECT \***

**FROM `student`**

**WHERE dob < '1975-01-01'**



**Question 2:**

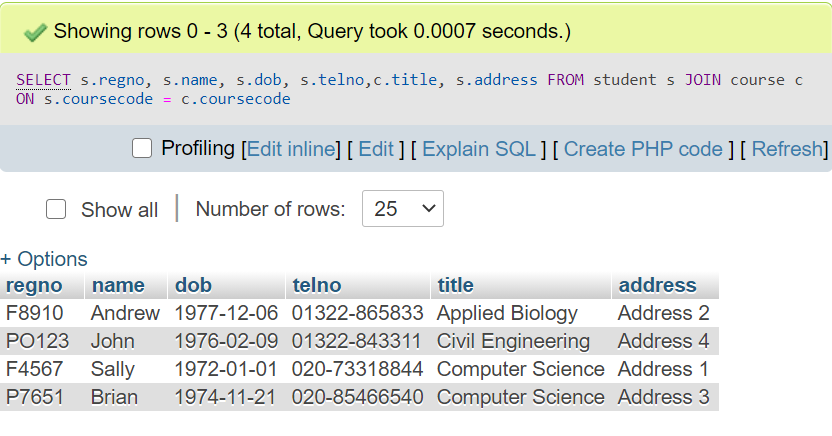
List all students’ details with the title of their course.

**SELECT s.regno, s.name, s.dob, s.telno,c.title, s.address**

**FROM student s**

**JOIN course c**

**ON s.coursecode = c.coursecode**



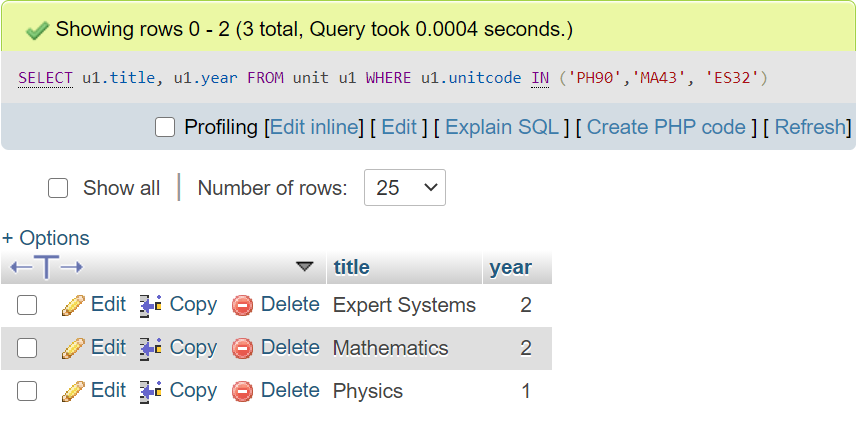
**Question 3:**

List the year of the following units: *Physics*, *Mathematics*, *Expert Systems*.

**SELECT title, year**

**FROM unit**

**WHERE unitcode IN ('PH90','MA43', 'ES32')**



**\*\*Here, the unit codes are stated within single inverted commas.**

**In the table structure of the table unit, it can be identified that the attribute ‘unitcode’ is created as a Varchar data type….**

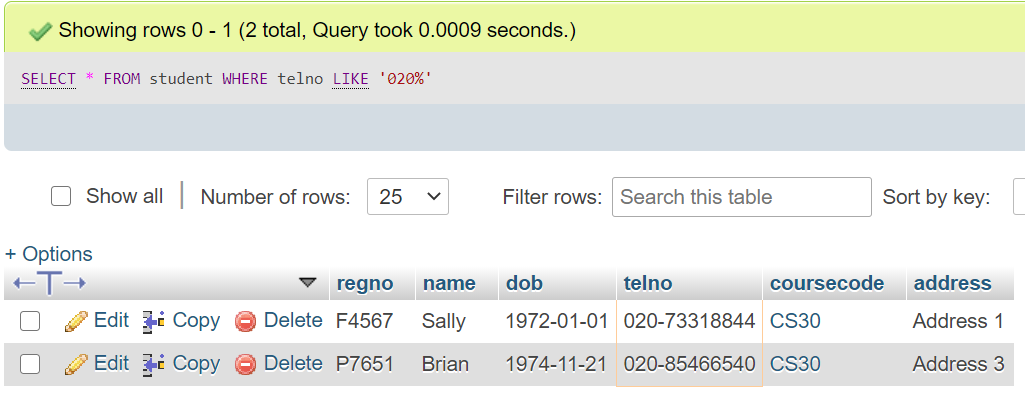
**Question 4:**

In order to find out which students live in London, list all students whose telephone number begins with ‘*020* ’.

**SELECT \***

**FROM student**

**WHERE telno LIKE '020%'**



**Question 5: (Question is not clear to me)**

For every student who has taken exams, list the average of their exam marks provided the average is more than 50.

**SELECT result.regno, AVG(exammark) AS avgmark**

**FROM result, student**

**WHERE result.regno = student.regno**

**GROUP BY regno**

**HAVING AVG(result.exammark)>50**

**OR**

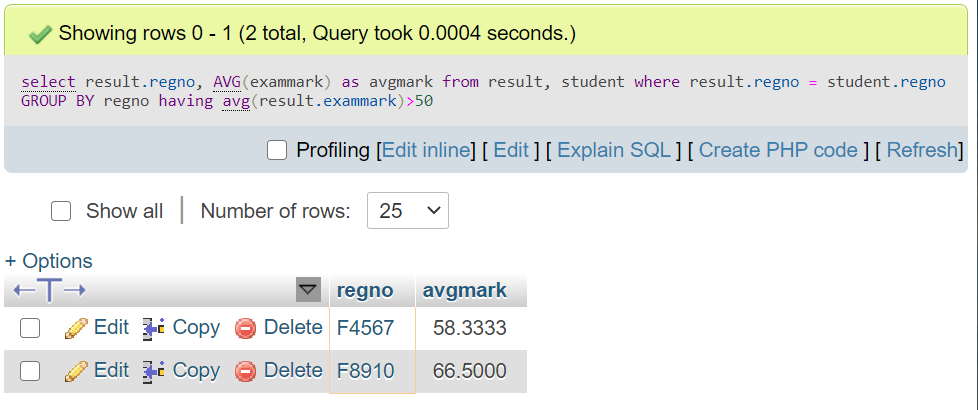
**SELECT result.regno,student.name, AVG(exammark) AS avgmark**

**FROM result, student**

**WHERE result.regno = student.regno**

**GROUP BY regno**

**HAVING AVG(result.exammark)>50**



**Question 6:**

List the students who have taken the following units: *Food Preservation* and *Relational Databases*.

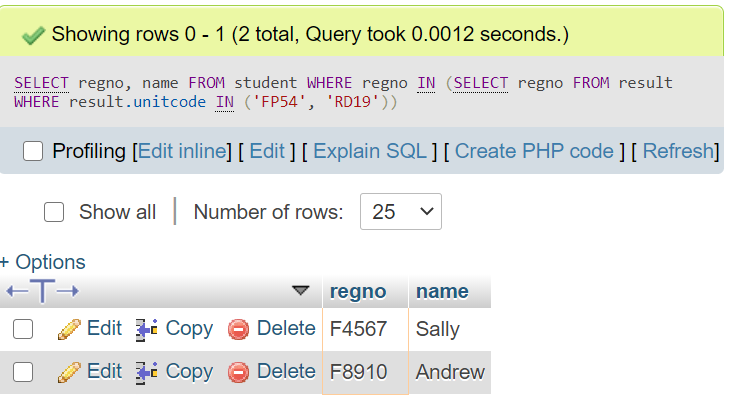
**SELECT regno, name**

**FROM student**

**WHERE regno IN (SELECT regno**

**FROM result**

**WHERE result.unitcode IN ('FP54', 'RD19'))**



**Question 7:**

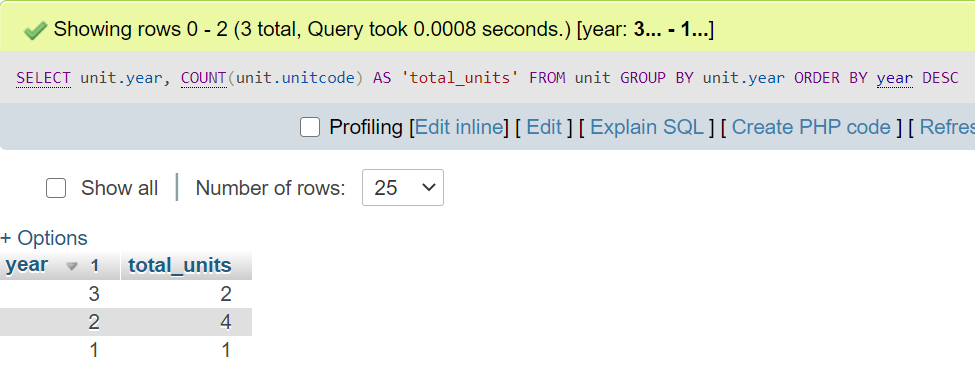
Count the number of units offered at each academic stage (year). The attributes required are *year* and *total\_units*. List the output in descending order of *year*.

**SELECT year, COUNT(unitcode) AS 'total\_units'**

**FROM unit**

**GROUP BY year**

**ORDER BY year DESC**

****

**Question 8:**

Give the name of the unit where the highest coursework mark was obtained.

**SELECT title, unit.unitcode**

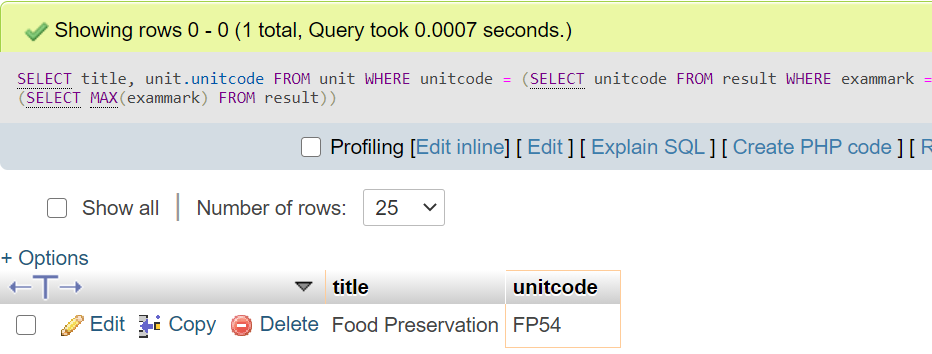
**FROM unit**

**WHERE unitcode = (SELECT unitcode**

**FROM result**

**WHERE exammark = (SELECT MAX(exammark)**

**FROM result))**



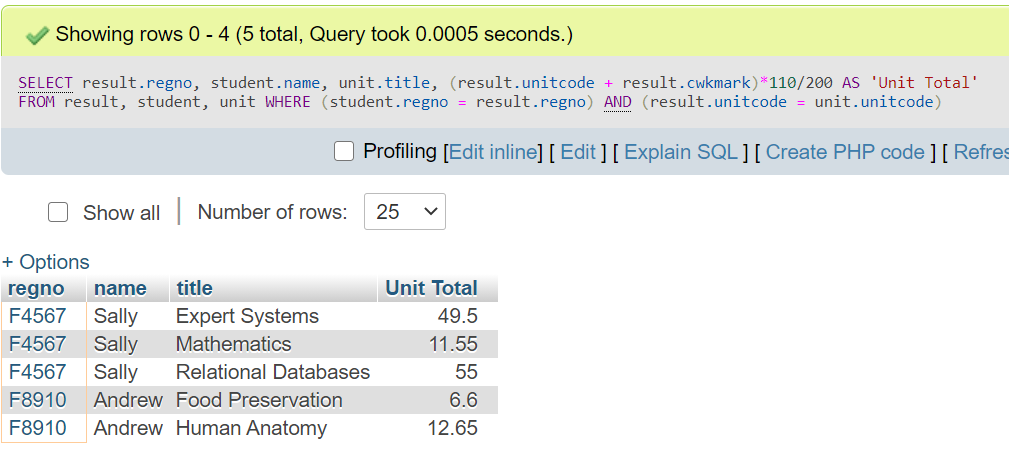
**Question 9:**

List the name, unit title and unit total (average of exam and coursework marks) of each student, having increased their unit total by 10%.

**SELECT result.regno, student.name, unit.title, (result.unitcode + result.cwkmark)\*110/200 AS 'Unit Total'**

**FROM result, student, unit**

**WHERE (student.regno = result.regno) AND (result.unitcode = unit.unitcode)**



**Question 10:**

Give the highest and lowest coursework mark for each unit. In the output table, both the unit code and the title of the unit are required.

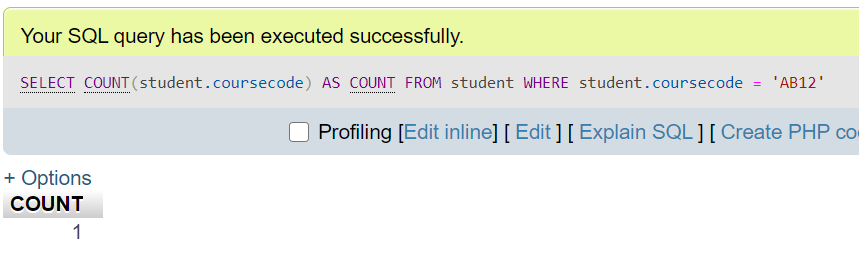
**Question 11:**

Find out how many students belong to the ‘*School of Life Sciences*’ without listing them.

**SELECT COUNT(student.coursecode) AS COUNT**

**FROM student**

**WHERE student.coursecode = 'AB12'**



Here…. From the course table, we can find that the ‘coursecode’ for Life Sciences school is AB12. So, in the student table the count having AB12 as the course code is taken

You can also use a subquery for this….

**SELECT COUNT(student.coursecode) AS COUNT**

**FROM student**

**WHERE student.coursecode = (SELECT course.coursecode**

**FROM course**

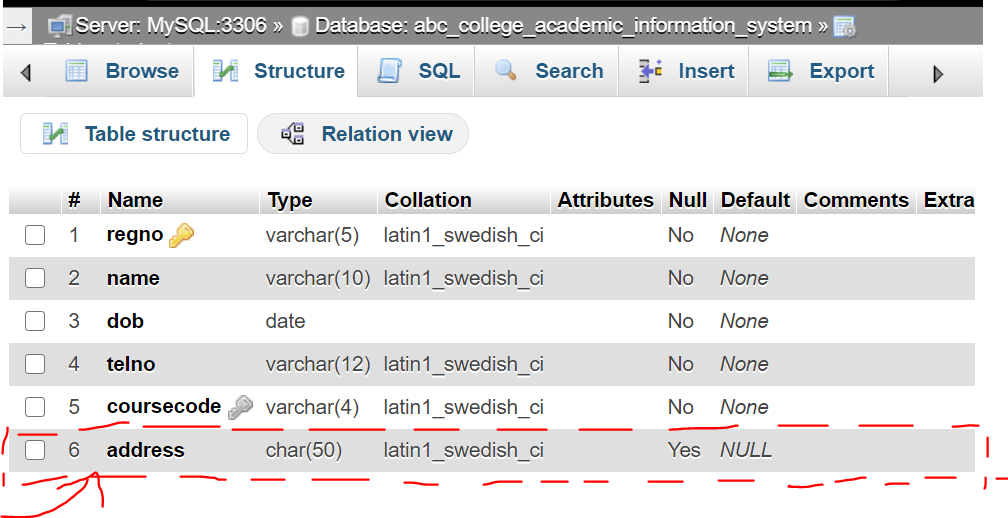
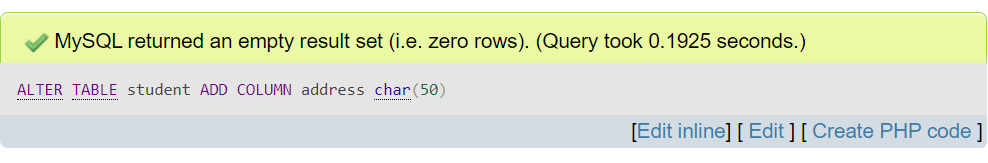
**WHERE course.school = 'Life Sciences')**

**Question 12:**

Add an attribute called ‘*address*’ to the *STUDENT* table.

Enter an address for each student.

**ALTER TABLE student**

**ADD COLUMN address char(50)**

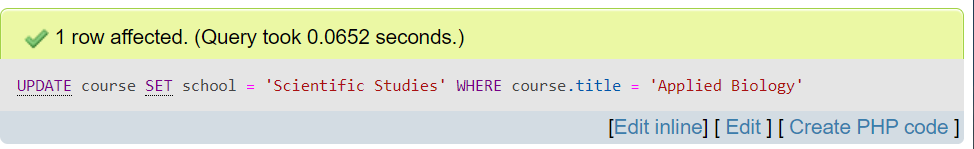
**Question 13:**

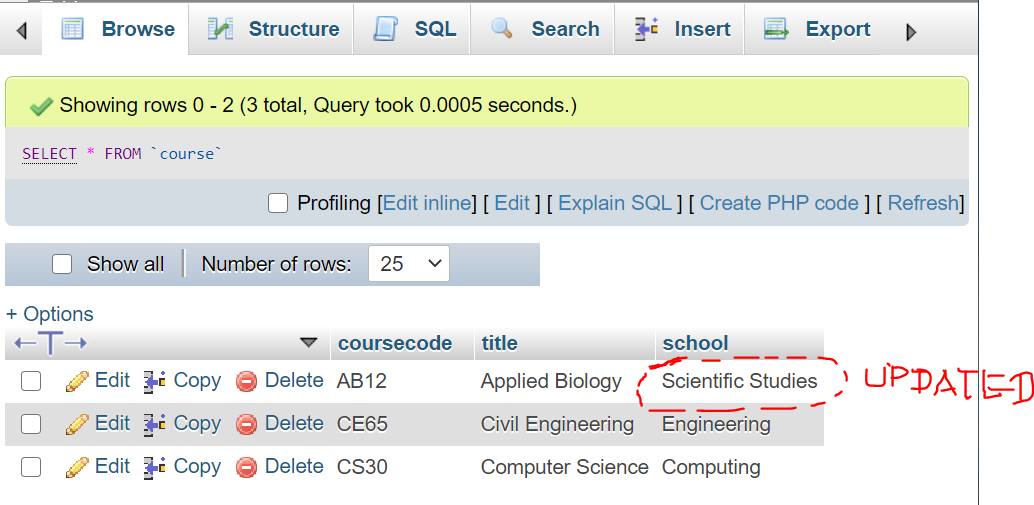
Write an update query to change the name of the school that offers ‘*Applied Biology*’ from *Life Sciences* to *Scientific Studies*.

**UPDATE course**

**SET school = 'Scientific Studies'**

**WHERE title = 'Applied Biology'**





**Question 14:**

Write an update query to delete all records related to the unit code *ES32* (from the result table) where the average of the exam and coursework marks (for the particular record) is less than 50.