KEYU CHEN – km5ar

The Question 3.12 and 3.12 is based on the version 7 of the textbook

3.11(a) a. Find the ID and name of each student who has taken at least one Comp. Sci. course; make sure there are no duplicate names in the result.

SELECT DISTINCT ID, name

FROM student

NATURAL JOIN takes

WHERE dept_name ='Comp. Sci.';

b. Find the ID and name of each student who has not taken any course offered before 2017.

SELECT DISTINCT ID, name

FROM student

NATURAL JOIN takes

WHERE ID NOT IN (**SELECT DISTINCT** ID

FROM STUDENT

NATURAL JOIN takes

WHERE year <2017);

c) For each department, find the maximum salary of instructors in that department. You may assume that every department has at least one instructor.

SELECT dept_name, MAX(salary) **AS** max_salary

FROM instructor

GROUP BY dept_name ;

d) Find the lowest, across all departments, of the per-department maximum salary computed by the preceding query.

SELECT MIN(max_salary)

FROM (**SELECT** dept_name, **MAX**(salary) AS max_salary

FROM instructor

GROUP BY dept_name) ;

3.12 Write the following queries in SQL , using the university schema.

a. Create a new course “CS-001”, titled “Weekly Seminar”, with 0 credits.

INSERT INTO course

VALUES ('CS-001', 'Weekly Seminar', 'Comp. Sci.', 0);

b. Create a section of this course in Fall 2017, with sec id of 1, and with the location of this section not yet specified.

INSERT INTO section (course_id, sec_id, semester, year, building)

VALUES ('CS-001', 1,'Fall', 2017, null);

c. Enroll every student in the Comp. Sci. department in the above section.

INSERT INTO takes (ID, course_id, sec_id, semester, year)

SELECT ID, 'CS-001', '1', 'Fall', 2017

FROM student

WHERE course_id = 'CS-001' ;

d. Delete enrollments in the above section where the student's ID is 12345.

DELETE FROM takes

WHERE ID= 12345;

f. Delete all takes tuples corresponding to any section of any course with the word "advanced" as a part of the title; ignore case when matching the word with the title.

DELETE FROM takes

WHERE course_id **IN**

(**SELECT** course_id **FROM** course

WHERE lower (title)

LIKE '%advanced%') ;

3.13

```
CREATE TABLE person (
driver_ID INT(20),
name VARCHAR(30) NOT NULL,
address VARCHAR(50)
PRIMARY KEY (driver_ID));

CREATE TABLE car (
license CHAR(20) PRIMARY KEY,
model VARCHAR(20) NOT NULL,
year INT(4));

CREATE TABLE accident(
report_number INT(15),
date DATE,
location VARCHAR(50)
PRIMARY KEY(report_number));

CREATE TABLE owns(
driver_ID INT(20) REFERENCES person,
```

```
license CHAR(20) REFERENCES car,
PRIMARY KEY(driver_ID));

CREATE TABLE participated (
report_number INT(15),
license CHAR(20) REFERENCES car,
driver_ID INT(20)REFERENCES person,
damage_amount INT
PRIMARY KEY (report_number)) ;
```