SQL Triggers Lab

Using the courses data (read the files `courses-ddl.sql and courses-small.sql`):

1. In SQLite, add a 'salary_audits' table, as defined in lecture.

```
create table salary_audits (
   ID   varchar(5),
   name varchar(20),
   dept_name varchar(20),
   old_salary numeric(8,2),
   new_salary numeric(8,2),
   time_of varchar(25)
);
```

2. Add a 'salary_update' trigger, that will cause a salary update to be audited if the salary increase is more than 10%:

```
create trigger salary_update after update on instructor
  when (new.salary > old.salary * 1.1)
begin
   insert into salary_audits values(new.ID, new.name, new.dept_name,
                                    old.salary, new.salary, datetime('now'));
 end;
```

3. Test the trigger by updating two instructor salaries. First, try increasing one instructor's salary by a small amount (maybe about 4%). Next, try increasing an instructor's salary by a large amount (say around 20%). Check to see the trigger does what it is supposed to.

   (Hint: can you remember the form of a an SQL update statement? It typically has the form `update <table> set … where …;`).

4. Repeat steps 1-3, but make a 'total_credits_audit' table, to provide a record of students who are signing up for more than 16 credits in a semester. Decide for yourself what the columns of the new table should be. Add a trigger that will cause a insert into the takes table to be audited if a student's total credits become more than 16.

   (Hint: it makes things easier if you define a 'student_courses' view, having the associated query "`select ID, course_id, semester, year, credits from takes natural join course;`").