RAWDATA Assignment 2 – Querying IMDB with SQL

This assignment concerns development of functions and procedures that extract info from the version of the IMDB database available on the wt-220.ruc.dk server.

Use your account on wt-220.ruc.dk

You are all created as users on wt-220.ruc.dk. Your have received a password in your ruc.dk mail. With your account at wt-220.ruc.dk you will have all rights to your own database (database-name = your user-name) as well as to group database "raw<your group number>". The functions and procedures you are supposed to develop should access the "imdb_movie"-database and you can find a copy of this database on the wt-220.ruc.dk-server as well. You are granted select-rights on all tables in this database – but only select. However, what you can do is the following. Create each function/procedure in your group database (raw1, raw2, raw3, raw4 or raw5) and access imdb_movie by using imdb_movie as a prefix. (If you want to try first individually you can do the same with your personal database). The appendix in this assignment text includes an example of a simple script that does just that (replace troels with your own user-name).

How and when to hand in

Generate the two text files described in the appendix on page 3 (that is, your SQL script and the result from running the SQL script using the "Execute (All or Selection) to text"). Upload these text-files to Moodle "Assignment 2" no later than February 22.

Hand in one submission from your group (from one of the members). Leave also your defined functions and procedures in your group database on the wt-220.ruc.dk server.

Question a)

The following SQL-query counts the numbers of movies Kevin Bacon has participated in.

```
SELECT count(distinct movie_id)
FROM imdb_movie.casting c, imdb_movie.person p,
imdb_movie.movie m
WHERE c.person_id = p.id
        AND c.movie_id = m.id
        AND m.kind_id=1
        AND c.role_type_id=1
AND p.name like 'Bacon, Kevin';
```

Write a function in SQL, **`movie_count(actor_name)`**, that returns the number of movies the actor **`actor_name`**, has participated in. Thus

```
SELECT movie_count('Bacon, Kevin');
```

should return the same result as the query above.

Question b)

Write a procedure, **`movies(actor_name)`**, that returns the titles of movies the actor **`actor_name`**, has participated in. Thus

```
CALL movie_count_proc('Mikkelsen, Mads');
```

should return the list of titles that Mads Mikkelsen has acted in.

Question c)

Write a procedure that takes a string as input and find the 10 most recent movies with a title that match the string.

Question d)

The following SQL-query retrieves the roles that Kevin Bacon has participated in.

```
SELECT DISTINCT role
FROM imdb_movie.casting
       JOIN imdb_movie.person
          ON person_id = person.id
       JOIN imdb_movie.role_type
          ON role_type_id = role_type.id
WHERE name like 'Bacon, Kevin';
```

Write a function in SQL, `roles(actor_name)`, that returns a comma-separated string listing the roles that Kevin Bacon has had. The function call

```
SELECT roles('Bacon, Kevin');
```

should return the following:

roles('Bacon, Kevin')
actor, producer, director, writer, editor, cinematographer

while the query

```
SELECT name,roles(name)
FROM imdb.name where name like 'De Niro, R%';
```

Should return the following

name	roles(name)
De Niro, Raphael	actor, producer
De Niro, Robert	actor, director, producer
De Niro, Rocco	actor
de Niro, Ryan	cinematographer

Hint: Use a cursor and loop through the query-result to assemble the string. The concat-function can be used for this purpose.

Appendix: Example SQL script and testing output

You are supposed to hand in two text files. The first text file must be a SQL script that define your functions and procedures and tests these. The second text file must be the result from running the SQL script using the "Execute (All or Selection) to text" in the Query-menu in MySQL Workbench. Example of both are given below.

SQL script - definition and test

Example SQL script that defines two procedures and a function and tests these.

```
use troels;
drop procedure if exists myfirst;
delimiter //
create procedure myfirst ()
begin
    select count(*)
    from imdb_movie.movie;
end;//
delimiter ;

drop procedure if exists mysecond;
delimiter //
create procedure mysecond ()
begin
    select distinct name
    from imdb_movie.person
    where name like 'Vargas, Fred%';
end;//
delimiter ;

drop function if exists hello;

delimiter //
create function hello (s char(20))
returns char(50)
begin
return concat('hello, ',s,'!');
end;//
delimiter ;


call myfirst();
call mysecond();
select hello(name) from university.instructor where
dept_name='Comp. Sci.';
```

Result from running the SQL script

Example text output from running the above SQL script using the "Execute (All or Selection) to text" in the Query-menu in MySQL Workbench.

```
Execute:
> call myfirst()

+ ------------- +
| count(*)      |
+ ------------- +
| 3570524       |
+ ------------- +
1 rows

Execute:
> call mysecond()

+ --------- +
| name      |
+ --------- +
| Vargas, Freddy |
| Vargas, Fred |
| Vargas, Fredy |
+ --------- +
3 rows

Execute:
> select hello(name) from university.instructor
where dept_name='Comp. Sci.'

+ ---------------- +
| hello(name)      |
+ ---------------- +
| hello, Srinivasan! |
| hello, Katz!     |
| hello, Brandt!   |
+ ---------------- +
3 rows
```