COMP6240 - Relational Databases

Assignment 1 (SQL)

Due date: 11:59pm, Aug 31, 2020

Instructions:

- **This assignment should be done individually (no group work**).
- This assignment will count for 15% of the final grade.
- A copy of the `moviedb` database is available on both the COMP2400 virtual machine (Option 1) and the ANU RSCS virtual environment (Option 2). You should first open a command shell and then connect to the `moviedb` database by entering

  ```
  psql moviedb
  ```

- You must submit one file: `myqueries.sql` for all the questions on Wattle before the due date. You can download the template files from the folder "Assignment 1 (SQL) for COMP6240" on Wattle. You must enter your queries into the template file, and more specifically, for the submitted file `myqueries.sql`, it should be executable in the given database `moviedb`

  ```
  moviedb=# \i myqueries.sql
  ```

- Late submission is not granted under any circumstance. You will be marked on whatever you have submitted at the time of the deadline. Please take careful note of deadlines and adhere to them. Of course, if you find yourself in a situation beyond your control that you believe significantly affects an assessment, you should follow the ANU's special consideration process (http://www.anu.edu.au/students/program-administration/assessments-exams/special-assessment-consideration).
- **Plagiarism will attract academic penalties in accordance with the ANU guidelines. A student in this course is expected to be able to explain and defend any submitted assessment item. The course convener can conduct or initiate an additional interview about any submitted assessment item for any student. If there is a significant discrepancy between the two forms of assessment, it will be automatically treated as a case of suspected academic misconduct.**

Question 1 **15 Marks**

The relational database `moviedb` has the following database schema:

MOVIE(title, production_year, country, run_time, major_genre)
`primary key` : {title, production_year}

PERSON(id, first_name, last_name, year_born)
`primary key` : {id}

AWARD(award_name, institution, country)
`primary key` : {award_name}

RESTRICTION_CATEGORY(description, country)
`primary key` : {description, country}

DIRECTOR(id, title, production_year)
`primary key` : {title, production_year}
`foreign keys` : [title, production_year] $\subseteq$ MOVIE[title, production_year]
[id] $\subseteq$ PERSON[id]

WRITER(id, title, production_year, credits)
`primary key` : {id, title, production_year}
`foreign keys` : [title, production_year] $\subseteq$ MOVIE[title, production_year]
[id] $\subseteq$ PERSON[id]

Crew(id, title, production_year, contribution)
`primary key` : {id, title, production_year}
`foreign keys` : [title, production_year] ⊆ Movie[title, production_year]
[id] ⊆ Person[id]

Scene(title, production_year, scene_no, description)
`primary key` : {title, production_year, scene_no}
`foreign keys` : [title, production_year] ⊆ Movie[title, production_year]

Role(id, title, production_year, description, credits)
`primary key` : {title, production_year, description}
`foreign keys` : [title, production_year] ⊆ Movie[title, production_year]
[id] ⊆ Person[id]

Restriction(title, production_year, description, country)
`primary key` : {title, production_year, description, country}
`foreign keys` : [title, production_year] ⊆ Movie[title, production_year]
[description, country] ⊆ Restriction_Category[description, country]

Appearance(title, production_year, description, scene_no)
`primary key` : {title, production_year, description, scene_no}
`foreign keys` : [title, production_year, scene_no]⊆Scene[title, production_year, scene_no]
[title, production_year, description]⊆Role[title, production_year, description]

Movie_Award(title, production_year, award_name, year_of_award,category, result)
`primary key` : {title, production_year, award_name, year_of_award, category}
`foreign keys` : [title, production_year] ⊆ Movie[title, production_year]
[award_name] ⊆ Award[award_name]

Crew_Award(id, title, production_year, award_name, year_of_award, category, result)
`primary key` : {id, title, production_year, award_name, year_of_award, category}
`foreign keys` : [id, title, production_year] ⊆ Crew[id, title, production_year]
[award_name] ⊆ Award[award_name]

Director_Award(title, production_year, award_name, year_of_award, category, result)
`primary key` : {title, production_year, award_name, year_of_award, category}
`foreign keys` : [title, production_year] ⊆ Director[title, production_year]
[award_name] ⊆ Award[award_name]

Writer_Award(id, title, production_year, award_name, year_of_award, category, result)
`primary key` : {id, title, production_year, award_name, year_of_award, category}
`foreign keys` : [id, title, production_year] ⊆ Writer[id, title, production_year]
[award_name] ⊆ Award[award_name]

Actor_Award(title, production_year, description, award_name, year_of_award,category,result)
`primary key` : {title, production_year, description, award_name, year_of_award, category}
`foreign keys` : [award_name] ⊆ Award[award_name]
[title,production_year,description]⊆Role[title,production_year,description]

There are five different categories of awards: movie awards, crew awards, director awards, writer awards and actor awards. A movie can only win an award after being nominated for the award.

Your task is to answer the following questions using SQL queries. For each question, your answer must be a *single SQL query* that may contain subqueries, and you must write and save your queries into the template file `myqueries.sql`..

1.1 How many *comedy* movies (i.e., the major genre of the movie is comedy) were produced in 1994? List that number. (1.5 Mark)

1.2 How many directors have directed at least one movie written by themselves? List that number. (1.5 Mark)

1.3 Who played two or more roles in the same movie? List their ids, the titles and production years of the corresponding movies. (1.5 Mark)

1.4 Which movies had the 'PG' restriction in at least two countries? List their titles, production years and the corresponding number of countries with the 'PG' restriction. (1.5 Mark)

1.5 Who have written exactly two American movies (*i.e.*, the production country is USA)? List their ids, first and last names. Order your results in the ascending order of their ids. (1.5 Mark)

1.6 How many directors have never played any roles in movies directed by themselves? List that number. (1.5 Mark)

1.7 Who won at least one director award and at least one writer award in the same year? List their ids and the corresponding year of award. (1.5 Mark)

1.8 Which crew member(s) worked on the greatest number of movies? List their id(s), first and last names. (1.5 Mark)

1.9 Who received the greatest number of nominations for a director award but never won? List their id(s). (1.5 Mark)

1.10 Find all the pairs of crew members who won a crew award at the same age. List the pairs of their ids. Note that the result should not contain duplicated pairs of ids, *e.g.*, (id_1, id_2) and (id_2, id_1) are considered as duplicated pairs and your query should only produce one of them in the result. Hint: if Emily (born in 1960) won a crew award in 1995 and Tom (born in 1955) won a crew award in 1990, they are considered as a pair of crew members who won a crew award at the same age (=35). (1.5 Mark)

+++++