# **Homework 5. Transition from RA to SQL**

Before you begin this homework, please watch 2 introductory SQL lectures of Jennifer Widom:

* Introduction to SQL: <https://www.youtube.com/watch?v=wxFmiRwXcQY>
* Basic select statement: <https://www.youtube.com/watch?v=4IxirOdp6bw>

## **Part 1. Queries on Movies dataset (submit through Markus as file HW5.pdf)**

Here is an all-familiar dataset about Movies.

**Movie** (**title**, **year**, **length**, **inColor**, **studioName**, **producerC**)

**MovieStar** (**name**, **address**, **gender**, **birthdate**)

**StarsIn** (**movieTitle**, **movieYear**, **starName**)

**MovieExec** (**name**, **address**, **certN**, **netWorth**)

**Studio** (**name**, **address**, **presCertN**)

You already have this set of tables implemented with PostgreSQL (see Homework 3). Now you can run moviesdata.sql to insert some basic data into your tables. Note that if some insertions violate the constraints you have set in Homework 3, you would need to disable these constraints for this exercise.

Translate following relational algebra queries into SQL and test your SQL against real Movies dataset.

Explain in plain English what each query is computing (1 sentence), provide SQL statement and an output you obtained after you run your query.

1. π name ( σ title = ‘Star wars’ (Movie) ⋈ producerC=certN MovieExec)

Find the name of the producer of Star Wars.

**SELECT name**

**FROM movie, "MovieExec"**

**WHERE title = 'Star Wars' AND producerc = certn;**

*name*

*--------------*

*George Lucas*

*(1 row)*

1. π title,name (Movie ⋈ producerC=certN MovieExec)

Find the title of the movie and the corresponding producer for the movie.

**SELECT title, name**

**FROM movie, "MovieExec"**

**WHERE producerc = certn;**

*title | name*

*-------------------------+-------------------*

*Titanic | James Cameron*

*Who Framed Roger Rabbit | Stephen Spielberg*

*Star Wars | George Lucas*

*Empire Strikes Back | George Lucas*

*(4 rows)*

1. πMovieStar.name, MovieExec.name(πname, address (MovieStar)

⋈MovieStar.address=MovieExec.address AND MovieStar.name !=MovieExec.name(πname, address (MovieExec)))

Find all Movie Stars and Movie Execs living at the same address where they cannot be the same person.

**SELECT "MovieStar".name, "MovieExec".name**

**FROM "MovieStar", "MovieExec"**

**WHERE "MovieStar".address = "MovieExec".address AND "MovieStar".name != "MovieExec".name;**

*name | name*

*------------+------------*

*Jane Fonda | Ted Turner*

*(1 row)*

1. πname((σtitle=‘Star wars’(Movie))⋈producerC!=certN(MovieExec))

Find all Movie Execs that did not produce Star Wars.

**SELECT name**

**FROM movie, "MovieExec"**

**WHERE title = 'Star Wars' AND producerc != certn;**

*name*

*-------------------*

*Melanie Griffith*

*Ted Turner*

*Stephen Spielberg*

*Merv Griffin*

*Calvin Coolidge*

*James Cameron*

*(6 rows)*

1. πname[(σstudioName=‘Disney’(Movie))⋈producerC=certN(MovieExec)]

∩ πname [(σstudioName=‘MGM’(Movie))⋈producerC=certN(MovieExec)]

Find all Movie Execs that produced both MGM and Disney Films.

**SELECT name**

**FROM movie, "MovieExec"**

**WHERE studioname = 'Disney' AND producerc = certn**

**INTERSECT**

**SELECT name**

**FROM movie, "MovieExec"**

**WHERE studioname = 'MGM' AND producerc = certn;**

*name*

*------*

*(0 rows)*

1. πtitle(Movie) – πtitle ((Movie)⋈producerC=certN(MovieExec))

Find all titles of the movies where the Movie Exec is not in the database.

**SELECT title**

**FROM movie**

**EXCEPT**

**SELECT title**

**FROM movie, "MovieExec"**

**WHERE producerc = certn;**

*title*

*-------------------------*

*The Usual Suspects*

*Pretty Woman*

*Gone with the wind*

*Star Trek: Nemesis*

*Mighty Ducks*

*Gone With the Wind*

*Godzilla*

*Wayne's world*

*Star Trek*

*Logan's run*

*The Man Who Wasnt There*

*Wyane*

*(12 rows)*

1. (ρS1(title1,year1,name1)(StarsIn)) ⋈name1=name2 AND (title1 != title2 or year1!=year2)( ρS2(title2,year2,name2)(StarsIn))

Find all movie stars that starred in at least 2 movies.

**SELECT s1.movietitle as title1, s1.movieyear as year1, s1.starname as name1, s2.movietitle as title2, s2.movieyear as year2, s2.starname as name2**

**FROM "StarsIn" s1, "StarsIn" s2**

**WHERE s1.starname = s2.starname AND (s1.movietitle != s2.movietitle OR s1.movieyear != s2.movieyear);**

*title1 | year1 | name1 | title2 | year2 | name2*

*---------------------+-------+---------------+---------------------+-------+---------------*

*Star Wars | 1977 | Harrison Ford | Empire Strikes Back | 1980 | Harrison Ford*

*Empire Strikes Back | 1980 | Harrison Ford | Star Wars | 1977 | Harrison Ford*

*(2 rows)*

## **Part 2. PCRS exercises**

The second part of this homework is on PCRS: <https://teach.cdf.toronto.edu/343/content/quests>

This is the first time we are using PCRS for this course, so if you encounter any problems please let me know right away.

There are total 7 Movies queries (7 points), 8 Countries queries (8 points), and 9 pizza queries (9 points), for a total of 24 points which still give you only 1% of the course grade, but you become proficient with SQL, which is the main topic of assignment 2.