

DBMS Assignment 3 - Lab 4 Report

Namisa Najah Raisa
Student ID: 210042112

September 5, 2023

1 Introduction

This is a report about the tasks given in the 4th lab.

2 Tasks

2.1 Task Overview

The lab consisted of writing SQL statements for 13 queries:

2.2 Statements

2.2.1 Writing the queries

```
1  --1. Find the name of the actors/actresses that are also
   directors (with and without set operator).
2
3  --without the set operator
4
5  SELECT DISTINCT ACT_FIRSTNAME, ACT_LASTNAME
6  FROM ACTOR
7  WHERE (ACT_FIRSTNAME || ' ' || ACT_LASTNAME) IN (SELECT
   DIR_FIRSTNAME || ' ' || DIR_LASTNAME
8  FROM DIRECTOR);
9
10 --with the set operator
11
```

```

12 SELECT NAME
13 FROM (
14 SELECT ACT_FIRSTNAME || ' ' || ACT_LASTNAME AS NAME FROM
    ACTOR
15 UNION ALL
16 SELECT DIR_FIRSTNAME || ' ' || DIR_LASTNAME AS NAME FROM
    DIRECTOR)
17 GROUP BY NAME
18 HAVING COUNT(*) > 1;

```

```

SQL>
SQL> --1
SQL>
SQL> --without set operator
SQL>
SQL> SELECT DISTINCT ACT_FIRSTNAME, ACT_LASTNAME
  2 FROM ACTOR
  3 WHERE (ACT_FIRSTNAME || ' ' || ACT_LASTNAME) IN (
  4     SELECT DIR_FIRSTNAME || ' ' || DIR_LASTNAME
  5     FROM DIRECTOR
  6 );

```

ACT_FIRSTNAME	ACT_LASTNAME
Woody	Allen
Orson	Welles
Kevin	Spacey

```

SQL>
SQL> --with set operator
SQL>
SQL> SELECT NAME
  2 FROM (
  3     SELECT ACT_FIRSTNAME || ' ' || ACT_LASTNAME AS NAME
  4     FROM ACTOR
  5     UNION ALL
  6     SELECT DIR_FIRSTNAME || ' ' || DIR_LASTNAME AS NAME
  7     FROM DIRECTOR
  8 )
  9 GROUP BY NAME
10 HAVING COUNT(*) > 1;

```

NAME
Woody Allen
Orson Welles
Kevin Spacey

```

1 --2.Find the actresses with the same first name.
2
3 SELECT ACT_FIRSTNAME, COUNT(*) AS NUM_ACTRESSES
4 FROM ACTOR
5 WHERE ACT_GENDER = 'F'
6 GROUP BY ACT_FIRSTNAME
7 HAVING COUNT(*) > 1;

```

```

SQL>
SQL> --2
SQL> SELECT ACT_FIRSTNAME, COUNT(*) AS NUM_ACTRESSES
      2 FROM ACTOR
      3 WHERE ACT_GENDER = 'F'
      4 GROUP BY ACT_FIRSTNAME
      5 HAVING COUNT(*) > 1;

```

ACT_FIRSTNAME	NUM_ACTRESSES
Kate	2
Jennifer	2

```

1 --3.Find the list of all the full names stored in the
  database.
2
3 SELECT ACT_FIRSTNAME || ' ' || ACT_LASTNAME AS FULL_NAME
4 FROM ACTOR
5 UNION
6 SELECT DIR_FIRSTNAME || ' ' || DIR_LASTNAME AS FULL_NAME
7 FROM DIRECTOR;

```

```

SQL> --3
SQL> SELECT ACT_FIRSTNAME || ' ' || ACT_LASTNAME AS FULL_NAME
2 FROM ACTOR
3 UNION
4 SELECT DIR_FIRSTNAME || ' ' || DIR_LASTNAME AS FULL_NAME
5 FROM DIRECTOR;

FULL_NAME
-----
Al Pacino
Alfred Hitchcock
Ali Astin
Andrei Tarkovsky
Bryan Singer
Christian Bale
Christopher Nolan
Claire Danes
Danny Boyle
David Aston
David Lean

FULL_NAME
-----
Deborah Kerr
Dev Patel
Eddie Redmayne
Ewan McGregor
F. Murray Abraham
Felicity Jones
Frank Darabont
George Raft
Gus Van Sant
Harrison Ford
Hayao Miyazaki

```

FULL_NAME

Jack Clayton
Jack Nicholson
Jackie Chan
James Cameron
James Marsh
James Stewart
Jennifer Aniston
Jennifer Garner
John Boorman
Jon Voight
Kate Mara

FULL_NAME

Kate Winslet
Kevin Spacey
Lana Condor
Maggie Gyllenhaal
Mark Wahlberg
Michael Cimino
Milos Forman
Nicole Kidman
Orson Welles
Paul Thomas Anderson
Peter Jackson

```
FULL_NAME
-----
Peter OToole
Raoul Walsh
Richard Kelly
Ridley Scott
Robert De Niro
Robert Duvall
Robin Williams
Roman Polanski
Sam Mendes
Shelley Duvall
Sigourney Weaver
```

```
FULL_NAME
-----
Stanley Kubrick
Stephen Baldwin
Susan Johnson
Tim Robbins
Woody Allen
```

```
60 rows selected.
```

```
1  --4.Find the movie titles that did not receive any ratings.
2
3  SELECT MOV_TITLE
4  FROM MOVIE
5  WHERE MOV_ID NOT IN (
6      SELECT DISTINCT MOV_ID
7      FROM RATING
8  );
```

```

SQL>
SQL> --4
SQL> SELECT MOV_TITLE
  2   FROM MOVIE
  3   WHERE MOV_ID NOT IN (
  4       SELECT DISTINCT MOV_ID
  5       FROM RATING
  6   );

```

MOV_TITLE

Citizen Kane
They Drive by Night
Spirited Away
To All the Boys I've Loved Before
The Theory of Everything
Back to the Future
Seven Samurai

7 rows selected.

```

1  --5. Find the average rating of all movies.
2
3  SELECT AVG(REV_STARS)
4  FROM RATING;

```

```

SQL>
SQL> --5
SQL> SELECT AVG(REV_STARS)
  2   FROM RATING;

```

AVG(REV_STARS)

6.8

```

1 --6.Find the minimum rating for each movie and display them
   in descending order of rating.
2
3 SELECT MOV_ID, MIN(REV_STARS) AS MIN_RATING
4 FROM RATING
5 GROUP BY MOV_ID
6 ORDER BY MIN_RATING DESC;

```

```

SQL>
SQL> --6
SQL> SELECT MOV_ID, MIN(REV_STARS) AS MIN_RATING
      2 FROM RATING
      3 GROUP BY MOV_ID
      4 ORDER BY MIN_RATING DESC;

```

MOV_ID	MIN_RATING
933	7
924	6
917	6
903	5
909	5
913	5
916	5
905	4
901	4
919	4
902	4

MOV_ID	MIN_RATING
904	4
914	3
923	3
920	3
921	3
918	3
925	3
906	3
915	3
910	3
907	3

MOV_ID	MIN_RATING
911	3
922	2
908	2
912	1

26 rows selected.

```

1  --7.Find the title of the movie having an average rev_star
   higher than the average rev_star of all the movies.
2
3  SELECT MOV_TITLE
4  FROM MOVIE
5  WHERE MOV_ID IN (
6  SELECT MOV_ID
7  FROM RATING
8  GROUP BY MOV_ID
9  HAVING AVG(REV_STARS) > (SELECT AVG(REV_STARS) FROM RATING));

```

```

SQL>
SQL> --7
SQL>
SQL> SELECT MOV_TITLE
      2 FROM MOVIE
      3 WHERE MOV_ID IN (
      4 SELECT MOV_ID
      5 FROM RATING
      6 GROUP BY MOV_ID
      7 HAVING AVG(REV_STARS) > (SELECT AVG(REV_STARS) FROM RATING));

MOV_TITLE
-----
Deliverance
Amadeus
The Prestige
The Innocents
Good Will Hunting
The Shining
The Shawshank Redemption
Chinatown
Titanic
Eyes Wide Shut
Annie Hall

MOV_TITLE
-----
Avatar
Braveheart

13 rows selected.

```

```

1  --8. Find the name of actors/actresses and the number of
   ratings received by the movies in which they played a role
   .
2
3  SELECT a.ACT_FIRSTNAME || ' ' || a.ACT_LASTNAME AS
      ACTOR_ACTRESS_NAME,
4  m.MOV_TITLE,
5  COUNT(r.MOV_ID) AS NUMBER_OF_RATINGS
6  FROM ACTOR a, CASTS c, MOVIE m, RATING r
7  WHERE a.ACT_ID = c.ACT_ID
8  AND c.MOV_ID = m.MOV_ID
9  AND m.MOV_ID = r.MOV_ID
10 GROUP BY a.ACT_FIRSTNAME, a.ACT_LASTNAME, m.MOV_TITLE
11 ORDER BY ACTOR_ACTRESS_NAME, NUMBER_OF_RATINGS DESC;

```

```

SQL>
SQL> --8
SQL>
SQL> SELECT a.ACT_FIRSTNAME || ' ' || a.ACT_LASTNAME AS ACTOR_ACTRESS_NAME,
2  m.MOV_TITLE,
3  COUNT(r.MOV_ID) AS NUMBER_OF_RATINGS
4  FROM ACTOR a, CASTS c, MOVIE m, RATING r
5  WHERE a.ACT_ID = c.ACT_ID
6  AND c.MOV_ID = m.MOV_ID
7  AND m.MOV_ID = r.MOV_ID
8  GROUP BY a.ACT_FIRSTNAME, a.ACT_LASTNAME, m.MOV_TITLE
9  ORDER BY ACTOR_ACTRESS_NAME, NUMBER_OF_RATINGS DESC;

```

ACTOR_ACTRESS_NAME	
MOV_TITLE	NUMBER_OF_RATINGS
Christian Bale	
The Prestige	9
Claire Danes	
Princess Mononoke	10
Deborah Kerr	
The Innocents	13

ACTOR_ACTRESS_NAME	
MOV_TITLE	NUMBER_OF_RATINGS
Dev Patel	
Slumdog Millionaire	10
Ewan McGregor	
Trainspotting	16
F. Murray Abraham	
Amadeus	13

ACTOR_ACTRESS_NAME	
MOV_TITLE	NUMBER_OF_RATINGS
Harrison Ford	
Blade Runner	13
Jack Nicholson	
Chinatown	11
James Stewart	
Vertigo	9

ACTOR_ACTRESS_NAME	
MOV_TITLE	NUMBER_OF_RATINGS
Jon Voight	
Deliverance	11
Kate Winslet	
Titanic	15
Kevin Spacey	
Beyond the Sea	10

ACTOR_ACTRESS_NAME	
MOV_TITLE	NUMBER_OF_RATINGS
Kevin Spacey	
American Beauty	10
Maggie Gyllenhaal	
Donnie Darko	14

```

Mark Wahlberg
Boogie Nights                                12

```

```

ACTOR_ACTRESS_NAME
-----

```

```

MOV_TITLE                                NUMBER_OF_RATINGS
-----

```

```

Nicole Kidman
Eyes Wide Shut                                13

```

```

Peter O'Toole
Lawrence of Arabia                            12

```

```

Robert De Niro
The Deer Hunter                             10

```

```

ACTOR_ACTRESS_NAME
-----

```

```

MOV_TITLE                                NUMBER_OF_RATINGS
-----

```

```

Robin Williams
Good Will Hunting                            15

```

```

Shelley Duvall
The Shining                                 10

```

```

Sigourney Weaver
Aliens                                       16

```

```

ACTOR_ACTRESS_NAME
-----

```

```

MOV_TITLE                                NUMBER_OF_RATINGS
-----

```

```

Stephen Baldwin
The Usual Suspects                            13

```

```

Tim Robbins
The Shawshank Redemption                     13

```

```

Woody Allen
Annie Hall                                  14

```

```

24 rows selected.

```

```

1  --9. Find the name of the director of the movie having the
2     highest average rev_star.
3  SELECT d.DIR_FIRSTNAME || ' ' || d.DIR_LASTNAME AS
4     Director_Name
5  FROM DIRECTOR d
6  JOIN DIRECTION di ON d.DIR_ID = di.DIR_ID
7  JOIN (

```

```

7 SELECT MOV_ID, AVG(REV_STARS) AS Avg_Rev_Star
8 FROM RATING
9 GROUP BY MOV_ID
10 HAVING AVG(REV_STARS) = (
11 SELECT MAX(Avg_Rev_Star)
12 FROM (
13 SELECT MOV_ID, AVG(REV_STARS) AS Avg_Rev_Star
14 FROM RATING
15 GROUP BY MOV_ID)))
16 r ON di.MOV_ID = r.MOV_ID;

```

```

SQL>
SQL> --9
SQL> SELECT d.DIR_FIRSTNAME || ' ' || d.DIR_LASTNAME AS Director_Name
2 FROM DIRECTOR d
3 JOIN DIRECTION di ON d.DIR_ID = di.DIR_ID
4 JOIN (
5 SELECT MOV_ID, AVG(REV_STARS) AS Avg_Rev_Star
6 FROM RATING
7 GROUP BY MOV_ID
8 HAVING AVG(REV_STARS) = (
9 SELECT MAX(Avg_Rev_Star)
10 FROM (
11 SELECT MOV_ID, AVG(REV_STARS) AS Avg_Rev_Star
12 FROM RATING
13 GROUP BY MOV_ID)))
14 r ON di.MOV_ID = r.MOV_ID;

DIRECTOR_NAME
-----
Stanley Kubrick

```

```

1 --10.Find all the movie-related information of movies acted
   and directed by the same person.
2
3 SELECT m.MOV_ID, m.MOV_TITLE, m.MOV_YEAR, m.MOV_TIME, m.
   MOV_LANGUAGE, m.MOV_RELEASEDATE, m.MOV_COUNTRY
4 FROM MOVIE m, CASTS c, DIRECTION d
5 WHERE m.MOV_ID = c.MOV_ID
6 AND m.MOV_ID = d.MOV_ID
7 AND c.ACT_ID = d.DIR_ID;

```

```

SQL>
SQL> --10
SQL> SELECT m.MOV_ID, m.MOV_TITLE, m.MOV_YEAR, m.MOV_TIME, m.MOV_LANGUAGE, m.MOV_RELEASEDATE, m.MOV_COUNTRY
2 FROM MOVIE m, CASTS c, DIRECTION d
3 WHERE m.MOV_ID = c.MOV_ID
4 AND m.MOV_ID = d.MOV_ID
5 AND c.ACT_ID = d.DIR_ID;

no rows selected

```

```

1 --11.Find the title and average rating of the movies that
   have an average rev_star of more than 7.
2
3 SELECT m.MOV_TITLE, AVG(r.REV_STARS) AS AVERAGE_RATING
4 FROM MOVIE m, RATING r
5 WHERE m.MOV_ID = r.MOV_ID
6 GROUP BY m.MOV_TITLE
7 HAVING AVG(r.REV_STARS) > 7
8 ORDER BY AVERAGE_RATING DESC;

```

```

SQL>
SQL> --11
SQL> SELECT m.MOV_TITLE, AVG(r.REV_STARS) AS AVERAGE_RATING
      2 FROM MOVIE m, RATING r
      3 WHERE m.MOV_ID = r.MOV_ID
      4 GROUP BY m.MOV_TITLE
      5 HAVING AVG(r.REV_STARS) > 7
      6 ORDER BY AVERAGE_RATING DESC;

```

MOV_TITLE	AVERAGE_RATING
The Shining	8.4
The Shawshank Redemption	8.23076923
Braveheart	7.54545455
Good Will Hunting	7.42857143
Avatar	7.42857143
Chinatown	7.4
Deliverance	7.27272727
Eyes Wide Shut	7.15384615

8 rows selected.

```

1 --12.Find the reviewer who gives the highest number of lowest
   rev_star.
2
3 SELECT r.REV_NAME, COUNT(*) AS Lowest_Rev_Star_Count
4 FROM REVIEWER r, RATING ra
5 WHERE r.REV_ID = ra.REV_ID
6 AND ra.REV_STARS = (
7 SELECT MIN(REV_STARS)
8 FROM RATING)
9 GROUP BY r.REV_NAME
10 HAVING COUNT(*) = (
11 SELECT MAX(Count_Rev_Star)
12 FROM (
13 SELECT COUNT(*) AS Count_Rev_Star
14 FROM REVIEWER r, RATING ra
15 WHERE r.REV_ID = ra.REV_ID
16 AND ra.REV_STARS = (
17 SELECT MIN(REV_STARS)
18 FROM RATING)
19 GROUP BY r.REV_NAME));

```

```

SQL> --12
SQL> SELECT r.REV_NAME, COUNT(*) AS Lowest_Rev_Star_Count
  2 FROM REVIEWER r, RATING ra
  3 WHERE r.REV_ID = ra.REV_ID
  4 AND ra.REV_STARS = (
  5 SELECT MIN(REV_STARS)
  6 FROM RATING)
  7 GROUP BY r.REV_NAME
  8 HAVING COUNT(*) = (
  9 SELECT MAX(Count_Rev_Star)
 10 FROM (
 11 SELECT COUNT(*) AS Count_Rev_Star
 12 FROM REVIEWER r, RATING ra
 13 WHERE r.REV_ID = ra.REV_ID
 14 AND ra.REV_STARS = (
 15 SELECT MIN(REV_STARS)
 16 FROM RATING)
 17 GROUP BY r.REV_NAME));

```

REV_NAME	LOWEST_REV_STAR_COUNT
-----	-----
Scott LeBrun	1

```

1  --13. Find the name and average runtime of movies of different
    actors/actresses. Do not include any actor/actress who
    worked with James Cameron .
2
3  SELECT (a.ACT_FIRSTNAME || ' ' || a.ACT_LASTNAME) AS
    Actor_Actress_Name, AVG(m.MOV_TIME) AS Average_Runtime
4  FROM ACTOR a, CASTS c, MOVIE m
5  WHERE a.ACT_ID = c.ACT_ID
6  AND c.MOV_ID = m.MOV_ID
7  AND a.ACT_ID NOT IN (
8  SELECT DISTINCT c.ACT_ID
9  FROM CASTS c, DIRECTION d, DIRECTOR di
10 WHERE c.MOV_ID = d.MOV_ID
11 AND d.DIR_ID = di.DIR_ID
12 AND (di.DIR_FIRSTNAME || ' ' || di.DIR_LASTNAME) = 'James
    Cameron')
13 GROUP BY a.ACT_FIRSTNAME, a.ACT_LASTNAME;

```



```

SQL> --13
SQL> SELECT (a.ACT_FIRSTNAME || ' ' || a.ACT_LASTNAME) AS Actor_Actress_Name, AVG(m.MOV_TIME) AS Average_Runtime
2 FROM ACTOR a, CASTS c, MOVIE m
3 WHERE a.ACT_ID = c.ACT_ID
4 AND c.MOV_ID = m.MOV_ID
5 AND a.ACT_ID NOT IN (
6 SELECT DISTINCT c.ACT_ID
7 FROM CASTS c, DIRECTION d, DIRECTOR di
8 WHERE c.MOV_ID = d.MOV_ID
9 AND d.DIR_ID = di.DIR_ID
10 AND (di.DIR_FIRSTNAME || ' ' || di.DIR_LASTNAME) = 'James Cameron')
11 GROUP BY a.ACT_FIRSTNAME, a.ACT_LASTNAME;

```

ACTOR_ACTRESS_NAME	AVERAGE_RUNTIME
Woody Allen	93
Robin Williams	126
Christian Bale	130
Eddie Redmayne	123
Lana Condor	99
Orson Welles	119
Mark Wahlberg	155
George Raft	95
James Stewart	128
Harrison Ford	117
Nicole Kidman	159

ACTOR_ACTRESS_NAME	AVERAGE_RUNTIME
Claire Danes	134
Tim Robbins	142
Kevin Spacey	120
Dev Patel	120
Jon Voight	109
Felicity Jones	123
Maggie Gyllenhaal	113
Deborah Kerr	100
Ewan McGregor	94
Peter O'Toole	216
Jack Nicholson	130

ACTOR_ACTRESS_NAME	AVERAGE_RUNTIME
Robert De Niro	183
F. Murray Abraham	160
Stephen Baldwin	106
Shelley Duvall	146

26 rows selected.

3 Challenges

The challenge in this lab was to fully understand the query asked to be executed. While writing the statements it was getting very confusing to write the correct statement. Even if they weren't giving any errors there was no way to know if the statements were actually correct or not since there were too many records to look through manually to verify.