Assignment 2 SQL

Student name: Luo YiFeng Student ID: 20448589

Suppose you have started a new movie-rating website, and you have been collecting data on reviewers' ratings of various movies. There is not much data yet, but you can still try out some interesting queries. Here's the schema:

Movie (mID, title, year, director)

There is a movie with ID number mID, a title, a release year, and a director.

Reviewer (rID, name)

The reviewer with ID number *rID* has a certain *name*.

Rating (<u>rID</u>, <u>mID</u>, stars, ratingDate) [rID, mID are foreign keys. rID refers to Reviewer.rID, mID refers to Movie.mID]

The reviewer rID gave the movie mID a number of stars rating (1-5) on a certain ratingDate.

In order to check whether my answer is correct, I created a database with python. There are three tables in this database, and some basic data are inserted. The three tables and the inserted data are as below:

The code for how to create the database and insert the data has been uploaded to my github account:

mID	title	year	director
1	Ready Player One	2018	Steven Spielberg
2	The Post	2017	Steven Spielberg
3	The Lost World: Jurassic Park	1997	Steven Spielberg
4	Tenet	2020	Christopher Nolan
5	Dunkirk	2017	Christopher Nolan
6	Inception	2010	Christopher Nolan
7	Interstellar	2014	Christopher Nolan
8	Mulan	2020	Niki Caro
9	Cats	2019	Tom Hooper
10	Jack and Jill	2011	Adam Sandler
11	The Room	2003	Tommy Wiseau
12	dummy1	2000	dummy
13	dummy2	2000	dummy
14	Gone with the Wind	1952	Victor Fleming

Movies table

rID	mID	stars	ratingDate
4	1	5	2019
1	2	3	2018
3	3	4	2000
1	4	4	2020
2	5	4	2019
5	6	5	2018
1	7	5	2011
9	8	3	2020
9	9	1	2020
9	10	2	2012
7	11	1	NULL
6	12	NULL	NULL
6	14	NULL	NULL
7	14	2	2020
8	14	5	1990
9	14	5	2009
12	1	5	2020
12	3	5	2020
12	14	5	2020

Rating table

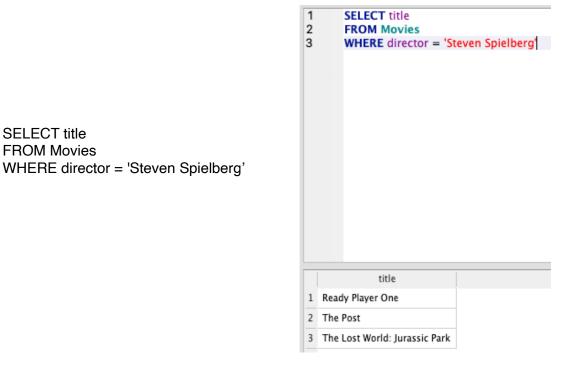
rID	name
1	Michael
2	Dwight
3	Pam
4	Jim
5	Andy
6	Jason
7	Andrewson
8	Andrew
9	Chris Jackson
10	Nobody
11	NO THIS PERSON
12	Sarah Martinez

Reviewer table

SELECT title **FROM Movies**

Write SQL statements to answer the following questions:

(1)(5') Find the titles of all movies directed by Steven Spielberg.



No.1

2(6') Find all years that have a movie that received a rating of 4 or 5, and sort them in increasing order of rating.

SELECT year FROM Movies, Rating WHERE Movies.mID = Rating.mID AND (stars = 4 or stars = 5) **ORDER BY stars**



Andrew Luo

3(6') Find the titles of all movies that have no ratings.

SELECT title
FROM Movies
a left join Rating b
on a.mID = b.mID
WHERE stars is null



No.4

(4) Some reviewers didn't provide a date with their rating. Find the names of all reviewers who have ratings with a NULL value for the date.

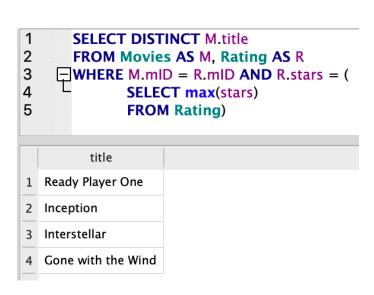
SELECT name

FROM Reviewer INNER JOIN Rating ON Reviewer.rID = Rating.rID WHERE ratingDate IS NULL

No.5

(5)(7') Find the title(s) of the movie(s) with the highest stars.

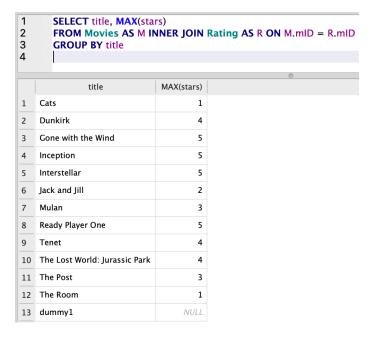
ELECT DISTINCT M.title
FROM Movies AS M, Rating AS R
WHERE M.mID = R.mID AND R.stars = (
SELECT max(stars)
FROM Rating)



NO.5

©(7') For each movie that has at least one rating, find the highest number of stars that movie received. Return the movie title and number of stars. Sort by movie title. (hint: use GROUP BY)

SELECT title, MAX(stars)
FROM Movies AS M INNER JOIN Rating AS R ON M.mID = R.mID
GROUP BY title



no.6

Andrew Luo

Andrew Luo

20448589 LuoYifeng

(7') <u>List movie titles and average ratings, from highest-rated to lowest-rated.</u> If two or more movies have the same average rating, list them in alphabetical order.

SELECT title, AVG(stars) AS avger

FROM Movies AS M INNER JOIN Rating AS R ON M.mID = R.mID

GROUP BY title

ORDER BY avger DESC, title



No.7

®(7') Find the names of all reviewers who rated Gone with the Wind.

SELECT DISTINCT name
FROM Movies
INNER JOIN Rating USING(mID)
INNER JOIN Reviewer USING(rID)
WHERE title = 'Gone with the Wind'

1 2 3 4 5	2 FROM Movies 3 INNER JOIN Rating USING(mID) 4 INNER JOIN Reviewer USING(rID)					
	name					
1	Jason					
2	Andrewson					
3	Andrew					
4	Chris Jackson					

No.8

20448589 LuoYifeng

(9(7') For any rating where the reviewer is the same as the director of the movie, return the reviewer name, movie title, and number of stars.

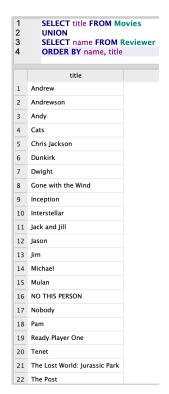
SELECT name, title, stars
FROM Movies
INNER JOIN Rating USING(mID)
INNER JOIN Reviewer USING(rID)
WHERE director = name

1 SELECT name, title, stars
2 FROM Movies
3 INNER JOIN Rating USING(mID)
4 INNER JOIN Reviewer USING(rID)
5 WHERE director = name

No.9

(D(7') Return all reviewer names and movie names together in a single column, alphabetized. (Sorting by the first name of the reviewer and first word in the title is fine; no need for special processing on last names or removing "The".)

SELECT title FROM Movies
UNION
SELECT name FROM Reviewer
ORDER BY name, title



No.10

(1)(7') Find the titles of all movies not reviewed by Chris Jackson.

```
SELECT title
FROM Movies
WHERE mID NOT IN (
SELECT mID
FROM Rating
INNER JOIN Reviewer USING(rID)
WHERE name = 'Chris Jackson'
)
```



No.11

(2) Find the movies reviewed by Sarah Martinez and not reviewed by Chris Jackson.

20448589 LuoYifeng



No.12

(3(7') <u>Find the movies that have been reviewed by at least two reviewers. Please</u> write an SQL with only one <u>SELECT</u> and without <u>GROUP BY.</u>

SELECT DISTINCT T1.title

FROM (Movies Join Rating USING (mID)) AS T1, (Movies Join Rating USING (mID)) AS T2

WHERE T1.mID = T2.mID AND T1.rID \Leftrightarrow T2.rID



No.13

(4)(7') Find the movies that have been reviewed by at least two reviewers. Please write an SQL with subquery.

```
SELECT title
FROM Movies
WHERE mID IN ( SELECT mID
FROM Rating R1
INNER JOIN Rating R2 USING (mID)
GROUP BY R1.mID
HAVING COUNT(*) >1)
```



No.14

(5(7') Find the reviewers who do not review any movie.

```
ELECT rID, name
FROM Reviewer
WHERE Reviewer.rID NOT IN ( SELECT DISTINCT Reviewer.rID FROM Reviewer, Rating WHERE Reviewer.rID = Rating.rID )
```

```
SELECT rID, name
2
      FROM Reviewer
3
    ■ WHERE Reviewer.rID NOT IN ( SELECT DISTINCT Reviewer.rID
4
                           FROM Reviewer, Rating
5
                           WHERE Reviewer.rID = Rating.rID
6
  rID
          name
1 10
      Nobody
     NO THIS PERSON
2 11
```

No.15