Database System Tutorial 2

You've started a new movie-rating website, and you've been collecting data on reviewers' ratings of various movies. There's not much data yet, but you can still try out some interesting queries.

Schema

You can download the database files from the following links:

- Download lab2.db – This is the database file. If you need a ready-to-use database, you can directly use this file.
- Download rating.sql – This SQL script contains the commands to create the database from scratch. If you prefer to generate the database yourself, use this script.

The database consists of the following tables:

- **Movie** (`mID`, `title`, `year`, `director`)
  Represents a movie with a unique ID (`mID`), a title, a release year, and a director.
- **Reviewer** (`rID`, `name`)
  Represents a reviewer with a unique ID (`rID`) and a name.
- **Rating** (`rID`, `mID`, `stars`, `ratingDate`)
  Represents a review where a reviewer (`rID`) gives a movie (`mID`) a star rating (1-5) on a specific date (`ratingDate`).

Task

Your task is to write SQL queries to retrieve information from a sample dataset based on this schema.

Instructions

For each problem, write an SQL query that answers the given question. Compare your query's output with the provided expected results to verify its correctness.

Important Notes

- Your queries are executed using SQLite, so you must conform to the SQL constructs supported by SQLite.
- Unless explicitly stated, you may return result rows in any order.
- You are to translate the English into a SQL query that computes the desired result over all possible databases. All you need to check is that your query gets the right answer on the small sample database. Thus, even if your output is correct, it is possible that your query does not correctly reflect the problem at hand. (For example, if we ask for a complex condition that requires accessing all of the tables, but over our small data set in the end the condition is satisfied only by Star Wars, then the query "select title from Movie where title = 'Star Wars'" will be correct even though it doesn't reflect the actual question.) Writing such queries may give the right answer on this dataset, but it does not help you properly learn SQL. On the other hand, if you attempt to write a general solution but make an error, your query will likely return incorrect results. Therefore, do not assume that a query is correct just because it produces the expected output on the small sample database. Always ensure that your query correctly reflects the given problem statement.

You are encouraged to experiment with and refine your queries to improve your understanding of SQL. Keep practicing until you are confident in your solutions!

Questions

1. Retrieve the titles of all movies where the director is Steven Spielberg.

```
SELECT title
FROM Movie
WHERE director = 'Steven Spielberg';
```

	title
1	E.T.
2	Raiders of the Lost Ark

2. Retrieve the years where there are one or more movies rated 4 or 5. Sort the years in ascending order.

```
SELECT DISTINCT year
FROM movie NATURAL JOIN rating
WHERE stars >= 4
ORDER BY year;
```

	year
1	1937
2	1939
3	1981
4	2009

3. Retrieve the titles of all movies that have never been rated.

```
SELECT title
FROM movie
WHERE mID NOT IN (SELECT mID FROM Rating);
```

	title
1	Star Wars
2	Titanic

4. Retrieve the names of all reviewers who submitted ratings WHERE the rating date is NULL.

```
SELECT name
FROM reviewer
WHERE rID IN (
    SELECT rID
    FROM Rating
    WHERE ratingDate is NULL);
```

	name
1	Daniel Lewis
2	Chris Jackson

5. Write a query to return the ratings data in a more readable format: reviewer name, movie title, stars, and ratingDate. Also, sort the data, first by reviewer name, then by movie title, and lastly by number of stars.

```
SELECT name, title, stars, ratingDate
FROM Movie NATURAL JOIN Reviewer NATURAL JOIN Rating
ORDER BY name, title, stars;
```

	name	title	stars	ratingDate
1	Ashley White	E.T.	3	2011-01-02
2	Brittany Harris	Raiders of the Lost Ark	2	2011-01-30
3	Brittany Harris	Raiders of the Lost Ark	4	2011-01-12
4	Brittany Harris	The Sound of Music	2	2011-01-20
5	Chris Jackson	E.T.	2	2011-01-22
6	Chris Jackson	Raiders of the Lost Ark	4	*NULL*
7	Chris Jackson	The Sound of Music	3	2011-01-27
8	Daniel Lewis	Snow White	4	*NULL*
9	Elizabeth Thomas	Avatar	3	2011-01-15
10	Elizabeth Thomas	Snow White	5	2011-01-19
11	James Cameron	Avatar	5	2011-01-20
12	Mike Anderson	Gone with the Wind	3	2011-01-09
13	Sarah Martinez	Gone with the Wind	2	2011-01-22
14	Sarah Martinez	Gone with the Wind	4	2011-01-27

6. Retrieve the names of reviewers and the titles of movies where the reviewer rated the same movie twice, with the second rating being higher than the first.

```
SELECT name, title
FROM (SELECT DISTINCT rid, mid
FROM (SELECT * FROM rating GROUP BY rid, mid
HAVING COUNT(*) = 2) AS r1 JOIN rating r2 USING (rid, mid)
WHERE (r1.stars < r2.stars) AND (r1.ratingDate < r2.ratingDate))
                                NATURAL JOIN movie
                                NATURAL JOIN reviewer;
```

	name	title
1	Sarah Martinez	Gone with the Wind

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.

```
SELECT title, score
FROM (SELECT MAX(stars) AS score, mid
    FROM rating
    GROUP BY mid) NATURAL JOIN movie
ORDER BY title;
```

	title	score
1	Avatar	5
2	E.T.	3
3	Gone with the Wind	4
4	Raiders of the Lost Ark	4
5	Snow White	5
6	The Sound of Music	3

8. Retrieve the titles of all movies along with their rating spread (the difference between the highest and lowest ratings). Sort results by rating spread (descending), then by title.

```
SELECT title, MAX(stars)-MIN(stars) AS rs
FROM movie NATURAL JOIN rating
GROUP BY mid
ORDER BY rs DESC, title;
```

	title	rs
1	Avatar	2
2	Gone with the Wind	2
3	Raiders of the Lost Ark	2
4	E.T.	1
5	Snow White	1
6	The Sound of Music	1

9. Compute the difference between the average per-movie rating of films released before 1980 and those released after 1980.

```
SELECT AVG(av1) - AVG(av2)
FROM (
        SELECT AVG(r1.stars) AS av1
        FROM rating r1 NATURAL JOIN movie
        WHERE year < 1980
        GROUP BY r1.mid
        ),

        (SELECT AVG(r2.stars) AS av2
            FROM rating r2 NATURAL JOIN movie
            WHERE year > 1980
            GROUP BY r2.mid
        )
;
```

	avg(av1) - avg(av2)
1	0.0555555555555558

10. Retrieve the names of all reviewers who submitted ratings for the movie *Gone with the Wind*.

```
SELECT DISTINCT name
FROM reviewer NATURAL JOIN rating
WHERE mid IN (SELECT m.mid FROM movie m WHERE m.title = 'Gone with the Wind');
```

	name
1	Sarah Martinez
2	Mike Anderson

11. 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 reviewer NATURAL JOIN rating NATURAL JOIN movie
WHERE name = director;
```

	name	title	stars
1	James Cameron	Avatar	5

12. Retrieve a combined list of reviewer names and movie titles, sorted alphabetically.

```
SELECT name AS r
FROM reviewer
UNION
SELECT title AS r
FROM movie
ORDER BY r;
```

	r
1	Ashley White
2	Avatar
3	Brittany Harris
4	Chris Jackson
5	Daniel Lewis
6	E.T.
7	Elizabeth Thomas
8	Gone with the Wind
9	James Cameron
10	Mike Anderson
11	Raiders of the Lost Ark
12	Sarah Martinez
13	Snow White
14	Star Wars
15	The Sound of Music
16	Titanic

13. Retrieve the titles of all movies that have not been reviewed by Chris Jackson.

```
SELECT title
FROM movie m
```

```
WHERE m.mid NOT IN (
    SELECT mid FROM rating NATURAL JOIN reviewer
    WHERE name = 'Chris Jackson'
    );
```

	title
1	Gone with the Wind
2	Star Wars
3	Titanic
4	Snow White
5	Avatar

14. Retrieve pairs of reviewer names where both reviewers rated the same movie. Eliminate duplicates, ensure that each pair appears only once, do not pair reviewers with themselves. For each pair, return the names in the pair in alphabetical order.

```
SELECT DISTINCT rev1.name, rev2.name
FROM rating r1 JOIN rating r2 JOIN reviewer rev1 JOIN reviewer rev2
ON (r1.rid = rev1.rid AND r2.rid = rev2.rid AND r1.mid = r2.mid)
WHERE rev1.name < rev2.name
ORDER BY rev1.name, rev2.name;
```

	name	name
1	Ashley White	Chris Jackson
2	Brittany Harris	Chris Jackson
3	Daniel Lewis	Elizabeth Thomas
4	Elizabeth Thomas	James Cameron
5	Mike Anderson	Sarah Martinez

15. For each rating that is the lowest (fewest stars) currently in the database, return the reviewer name, movie title, and number of stars.

```
SELECT DISTINCT name, title, stars
FROM rating rat NATURAL JOIN reviewer NATURAL JOIN movie
WHERE NOT EXISTS (SELECT r.stars FROM rating r
                  WHERE r.stars < rat.stars);
```

	name	title	stars
1	Sarah Martinez	Gone with the Wind	2
2	Brittany Harris	The Sound of Music	2
3	Brittany Harris	Raiders of the Lost Ark	2
4	Chris Jackson	E.T.	2

16. Retrieve movie titles along with their average ratings, sorted by average rating in descending order. If multiple movies share the same rating, sort them alphabetically.

```
SELECT title, AVG(stars) AS rat
FROM movie NATURAL JOIN rating
```

```
GROUP BY mid
ORDER BY rat DESC, title;
```

	title	rat
1	Snow White	4.5
2	Avatar	4.0
3	Raiders of the Lost Ark	3.33333333333333
4	Gone with the Wind	3.0
5	E.T.	2.5
6	The Sound of Music	2.5

17. Find the names of all reviewers who have contributed three or more ratings. (As an extra challenge, try writing the query without HAVING or without COUNT.)

```
SELECT name
FROM reviewer
WHERE reviewer.rid IN (SELECT r.rid
              FROM rating r, rating s, rating t
                WHERE r.rid = s.rid AND s.rid = t.rid AND
                    (r.mid,r.ratingdate) < (s.mid,s.ratingdate)
                     AND (s.mid, s.ratingdate) < (t.mid,t.ratingdate));
```

	name
1	Brittany Harris
2	Chris Jackson

18. Retrieve the names of directors who have directed more than one movie, along with the titles of all movies they directed. Sort by director name, then by movie title.

```
SELECT title, director
FROM movie
WHERE director IN (SELECT director FROM movie m1 JOIN movie m2
                   USING (director)
                   WHERE m1.mid <> m2.mid)
ORDER BY director, title;
```

	title	director
1	Avatar	James Cameron
2	Titanic	James Cameron
3	E.T.	Steven Spielberg
4	Raiders of the Lost Ark	Steven Spielberg

19. Find the movie(s) with the highest average rating. Return the movie title(s) and average rating.

```
SELECT title, x
FROM
    (SELECT AVG(stars) AS x, r.mid AS mid
    FROM rating r
    GROUP BY r.mid) JOIN movie USING (mid)
WHERE
NOT EXISTS
(SELECT AVG(stars) AS y FROM rating r GROUP BY r.mid HAVING x < y);
```

	title	x
1	Snow White	4.5

20. Find the movie(s) with the lowest average rating. Return the movie title(s) and average rating.

```
SELECT title, x
FROM
    (SELECT AVG(stars) AS x, r.mid AS mid
    FROM rating r
    GROUP BY r.mid) JOIN movie USING (mid)
WHERE
NOT EXISTS
(SELECT AVG(stars) AS y FROM rating r GROUP BY r.mid HAVING x > y);
```

	title	x
1	The Sound of Music	2.5
2	E.T.	2.5

21. For each director, return the director's name together with the title(s) of the movie(s) they directed that received the highest rating among all of their movies, and the value of that rating. Ignore movies whose director is NULL.

```
SELECT DISTINCT director, title, stars
FROM movie NATURAL JOIN rating
WHERE stars =
    (SELECT MAX(stars)
    FROM movie mov NATURAL JOIN rating
    WHERE mov.director = movie.director);
```

	director	title	stars
1	Victor Fleming	Gone with the Wind	4
2	Robert Wise	The Sound of Music	3
3	James Cameron	Avatar	5
4	Steven Spielberg	Raiders of the Lost Ark	4

Note: We just show the demonstration of before query execution and after query execution, this is not the actual final result.

22. Add the reviewer Roger Ebert to your database, with an `rID` of 209.

```
INSERT INTO reviewer VALUES (209, 'Roger Ebert');
```

	mID	title	year	director
1	101	Gone with the Wind	1939	Victor Fleming
2	102	Star Wars	1977	George Lucas
3	103	The Sound of Music	1965	Robert Wise
4	104	E.T.	1982	Steven Spielberg
5	105	Titanic	1997	James Cameron
6	106	Snow White	1937	*NULL*
7	107	Avatar	2009	James Cameron
8	108	Raiders of the Lost Ark	1981	Steven Spielberg

	mID	title	year	director
1	101	Gone with the Wind	1939	Victor Fleming
2	102	Star Wars	1977	George Lucas
3	103	The Sound of Music	1965	Robert Wise
4	104	E.T.	1982	Steven Spielberg
5	105	Titanic	1997	James Cameron
6	106	Snow White	1962	*NULL*
7	107	Avatar	2034	James Cameron
8	108	Raiders of the Lost Ark	1981	Steven Spielberg

23. For all movies that have an average rating of 4 stars or higher, add 25 to the release year. (Update the existing tuples; don't insert new tuples.)

```
UPDATE movie
SET year = movie.year + 25
WHERE movie.mid IN
    (SELECT rating.mid FROM rating GROUP BY rating.mid HAVING AVG(stars) >=4);
```

	mID	title	year	director
1	101	Gone with the Wind	1939	Victor Fleming
2	102	Star Wars	1977	George Lucas
3	103	The Sound of Music	1965	Robert Wise
4	104	E.T.	1982	Steven Spielberg
5	105	Titanic	1997	James Cameron
6	106	Snow White	1937	*NULL*
7	107	Avatar	2009	James Cameron
8	108	Raiders of the Lost Ark	1981	Steven Spielberg

	mID	title	year	director
1	101	Gone with the Wind	1939	Victor Fleming
2	102	Star Wars	1977	George Lucas
3	103	The Sound of Music	1965	Robert Wise
4	104	E.T.	1982	Steven Spielberg
5	105	Titanic	1997	James Cameron
6	106	Snow White	1962	*NULL*
7	107	Avatar	2034	James Cameron
8	108	Raiders of the Lost Ark	1981	Steven Spielberg

24. Delete all ratings where the movie's year is before 1970 or after 2000, and the rating is fewer than 4 stars.

```
DELETE FROM rating
WHERE rating.mid IN (
    SELECT movie.mid FROM movie
    WHERE year < 1970 OR year > 2000)
    AND rating.stars < 4;
```

	mID	title	year	director
1	101	Gone with the Wind	1939	Victor Fleming
2	102	Star Wars	1977	George Lucas
3	103	The Sound of Music	1965	Robert Wise
4	104	E.T.	1982	Steven Spielberg
5	105	Titanic	1997	James Cameron
6	106	Snow White	1937	*NULL*
7	107	Avatar	2009	James Cameron
8	108	Raiders of the Lost Ark	1981	Steven Spielberg

	mID	title	year	director
1	101	Gone with the Wind	1939	Victor Fleming
2	102	Star Wars	1977	George Lucas
3	103	The Sound of Music	1965	Robert Wise
4	104	E.T.	1982	Steven Spielberg
5	105	Titanic	1997	James Cameron
6	106	Snow White	1937	*NULL*
7	107	Avatar	2009	James Cameron
8	108	Raiders of the Lost Ark	1981	Steven Spielberg