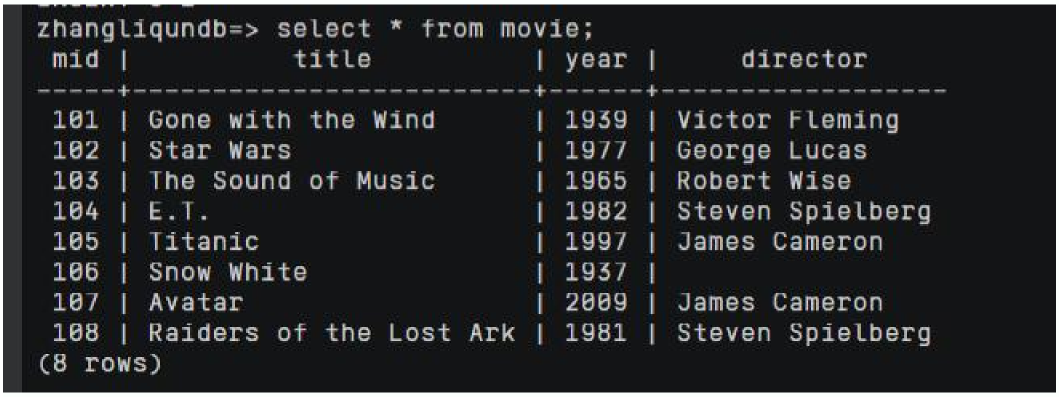
Movie (mID, title, year, director)
电影（电影编号，电影名，上映年份，导演）

Reviewer (rID, name)
评论者（评论者编号，评论者姓名）

Rating (rID, mID, stars, ratingDate)
评论（评论者编号，电影编号，评分，评论日期）
注： 评分范围 1-5 星；评论者在不同日期可以多次评价同一部电影。

Q1

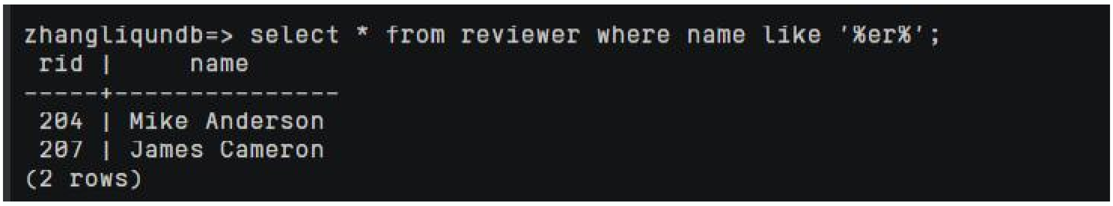
Find all movies.



```
zhangliqundb=> select * from movie;
 mid |          title          | year |     director
-----+-------------------------+------+------------------
 101 | Gone with the Wind      | 1939 | Victor Fleming
 102 | Star Wars               | 1977 | George Lucas
 103 | The Sound of Music      | 1965 | Robert Wise
 104 | E.T.                    | 1982 | Steven Spielberg
 105 | Titanic                 | 1997 | James Cameron
 106 | Snow White              | 1937 |
 107 | Avatar                  | 2009 | James Cameron
 108 | Raiders of the Lost Ark | 1981 | Steven Spielberg
(8 rows)
```

Q2

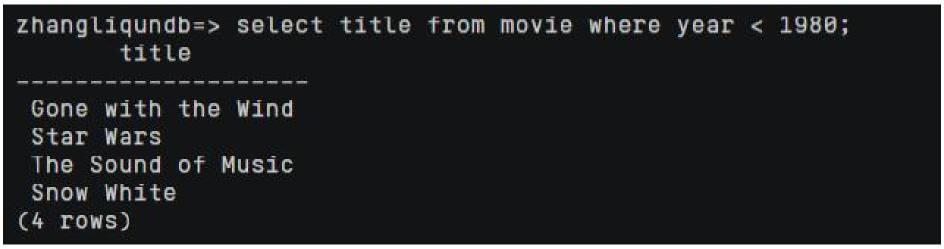
Find all reviewers whose name include ‘er’.



```
zhangliqundb=> select * from reviewer where name like '%er%';
 rid |     name
-----+---------------
 204 | Mike Anderson
 207 | James Cameron
(2 rows)
```

Q3

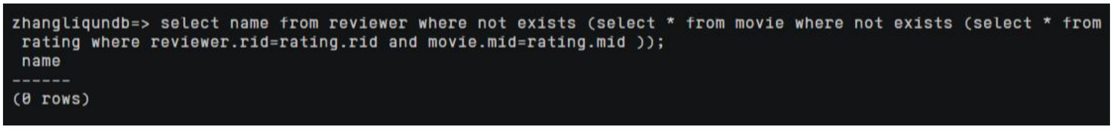
Find the titles of all movies released before 1980.



```
zhangliqundb=> select title from movie where year < 1980;
       title
--------------------
 Gone with the Wind
 Star Wars
 The Sound of Music
 Snow White
(4 rows)
```

Q4

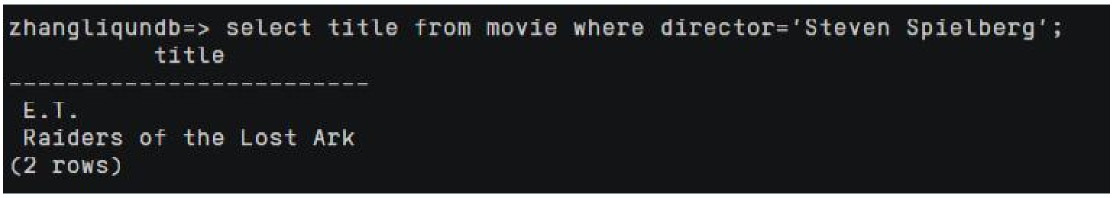
Find the reviewer names who have ratings with all movies.



```
zhangliqundb=> select name from reviewer where not exists (select * from movie where not exists (select * from
 rating where reviewer.rid=rating.rid and movie.mid=rating.mid ));
 name
------
(0 rows)
```

Q5

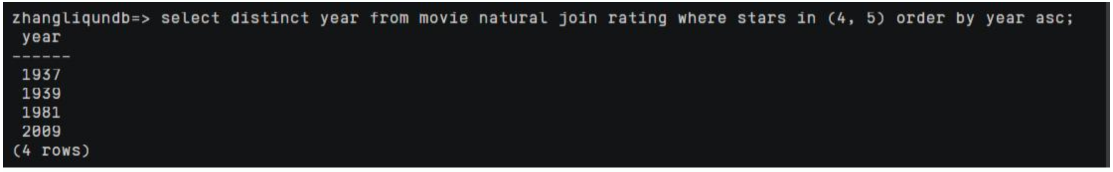
Find the titles of all movies directed by Steven Spielberg.



```
zhangliqundb=> select title from movie where director='Steven Spielberg';
          title
-------------------------
 E.T.
 Raiders of the Lost Ark
(2 rows)
```

Q6

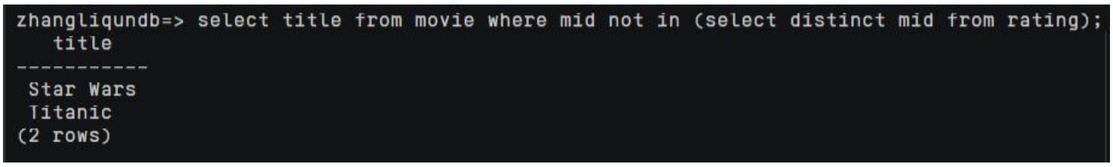
Find all years that have a movie that received a rating of 4 or 5, and sort them in increasing order.



```
zhangliqundb=> select distinct year from movie natural join rating where stars in (4, 5) order by year asc;
 year
------
 1937
 1939
 1981
 2009
(4 rows)
```

Q7

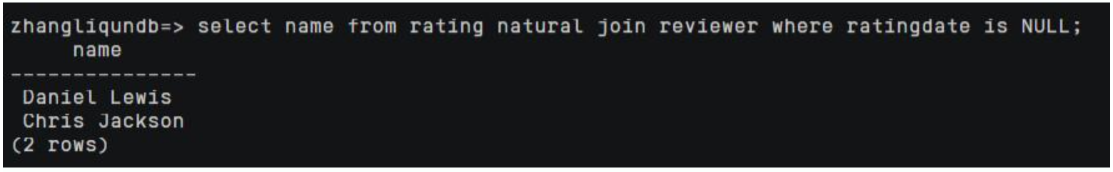
Find the titles of all movies that have no ratings.



```
zhangliqundb=> select title from movie where mid not in (select distinct mid from rating);
   title
-----------
 Star Wars
 Titanic
(2 rows)
```

Q8

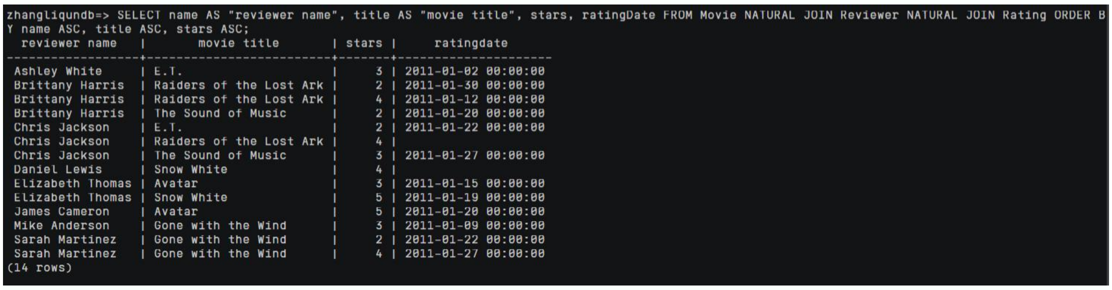
Some reviewers didn't provide a date with their rating. Find the names of all reviewers who have ratings with a NULL value for the date.



```
zhangliqundb=> select name from rating natural join reviewer where ratingdate is NULL;
     name
---------------
 Daniel Lewis
 Chris Jackson
(2 rows)
```

Q9

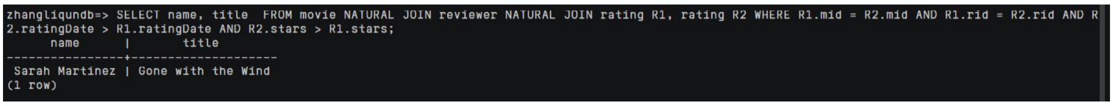
Write a query to return the ratings data in a more readable format: reviewer name, movie title, stars, and ratingDate. Also, sort the data, first by reviewer name, then by movie title, and lastly by number of stars.



```
zhangliqundb=> SELECT name AS "reviewer name", title AS "movie title", stars, ratingDate FROM Movie NATURAL JOIN Reviewer NATURAL JOIN Rating ORDER B
Y name ASC, title ASC, stars ASC;
  reviewer name   |       movie title       | stars |     ratingdate
------------------+-------------------------+-------+---------------------
 Ashley White     | E.T.                    |     3 | 2011-01-02 00:00:00
 Brittany Harris  | Raiders of the Lost Ark |     2 | 2011-01-30 00:00:00
 Brittany Harris  | Raiders of the Lost Ark |     4 | 2011-01-12 00:00:00
 Brittany Harris  | The Sound of Music      |     2 | 2011-01-20 00:00:00
 Chris Jackson    | E.T.                    |     2 | 2011-01-22 00:00:00
 Chris Jackson    | Raiders of the Lost Ark |     4 |
 Chris Jackson    | The Sound of Music      |     3 | 2011-01-27 00:00:00
 Daniel Lewis     | Snow White              |     4 |
 Elizabeth Thomas | Avatar                  |     3 | 2011-01-15 00:00:00
 Elizabeth Thomas | Snow White              |     5 | 2011-01-19 00:00:00
 James Cameron    | Avatar                  |     5 | 2011-01-20 00:00:00
 Mike Anderson    | Gone with the Wind      |     3 | 2011-01-09 00:00:00
 Sarah Martinez   | Gone with the Wind      |     2 | 2011-01-22 00:00:00
 Sarah Martinez   | Gone with the Wind      |     4 | 2011-01-27 00:00:00
(14 rows)
```

Q10

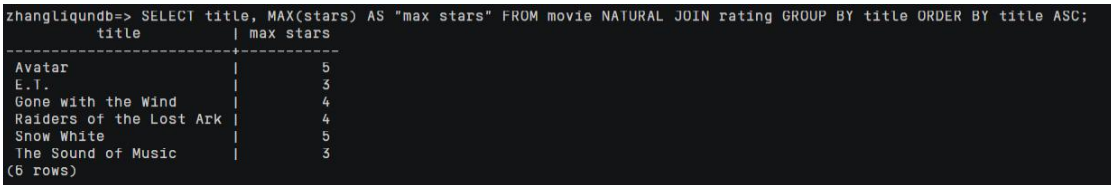
For all cases where the same reviewer rated the same movie twice and gave it a higher rating the second time, return the reviewer's name and the title of the movie.



```
zhangliqundb=> SELECT name, title  FROM movie NATURAL JOIN reviewer NATURAL JOIN rating R1, rating R2 WHERE R1.mid = R2.mid AND R1.rid = R2.rid AND R
2.ratingDate > R1.ratingDate AND R2.stars > R1.stars;
      name      |       title
----------------+--------------------
 Sarah Martinez | Gone with the Wind
(1 row)
```

Q11

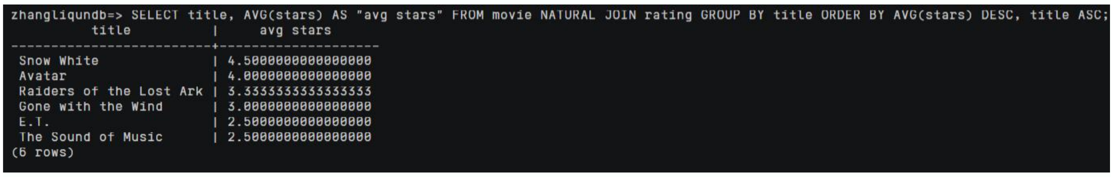
For each movie that has at least one rating, find the highest number of stars that movie received. Return the movie title and number of stars. Sort by movie title.



```
zhangliqundb=> SELECT title, MAX(stars) AS "max stars" FROM movie NATURAL JOIN rating GROUP BY title ORDER BY title ASC;
          title          | max stars
-------------------------+-----------
 Avatar                  |         5
 E.T.                    |         3
 Gone with the Wind      |         4
 Raiders of the Lost Ark |         4
 Snow White              |         5
 The Sound of Music      |         3
(6 rows)
```

Q12

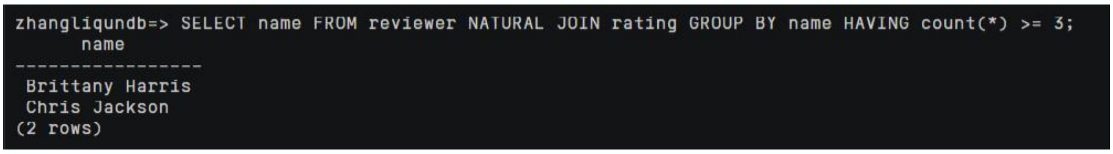
List movie titles and average ratings, from highest-rated to lowest-rated. If two or more movies have the same average rating, list them in alphabetical order.



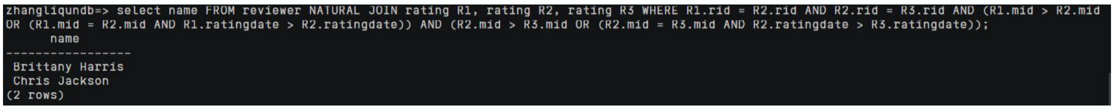
```
zhangliqundb=> SELECT title, AVG(stars) AS "avg stars" FROM movie NATURAL JOIN rating GROUP BY title ORDER BY AVG(stars) DESC, title ASC;
          title          |     avg stars
-------------------------+--------------------
 Snow White              | 4.5000000000000000
 Avatar                  | 4.0000000000000000
 Raiders of the Lost Ark | 3.3333333333333333
 Gone with the Wind      | 3.0000000000000000
 E.T.                    | 2.5000000000000000
 The Sound of Music      | 2.5000000000000000
(6 rows)
```

Q13

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.)



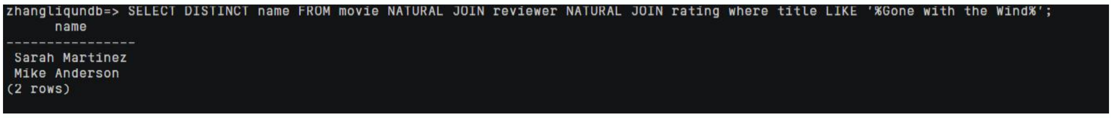
```
zhangliqundb=> SELECT name FROM reviewer NATURAL JOIN rating GROUP BY name HAVING count(*) >= 3;
      name
-----------------
 Brittany Harris
 Chris Jackson
(2 rows)
```



```
zhangliqundb=> select name FROM reviewer NATURAL JOIN rating R1, rating R2, rating R3 WHERE R1.rid = R2.rid AND R2.rid = R3.rid AND (R1.mid > R2.mid
OR (R1.mid = R2.mid AND R1.ratingdate > R2.ratingdate)) AND (R2.mid > R3.mid OR (R2.mid = R3.mid AND R2.ratingdate > R3.ratingdate));
      name
-----------------
 Brittany Harris
 Chris Jackson
(2 rows)
```

Q14

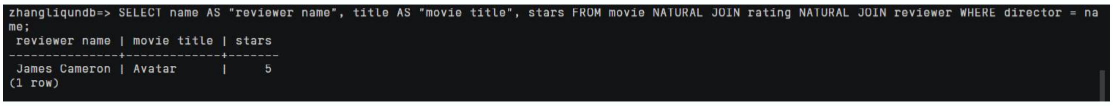
Find the names of all reviewers who rated Gone with the Wind.



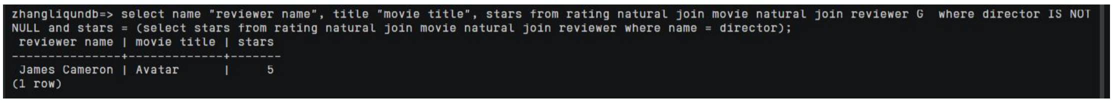
```
zhangliqundb=> SELECT DISTINCT name FROM movie NATURAL JOIN reviewer NATURAL JOIN rating where title LIKE '%Gone with the Wind%';
      name
-----------------
 Sarah Martinez
 Mike Anderson
(2 rows)
```

Q15

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.



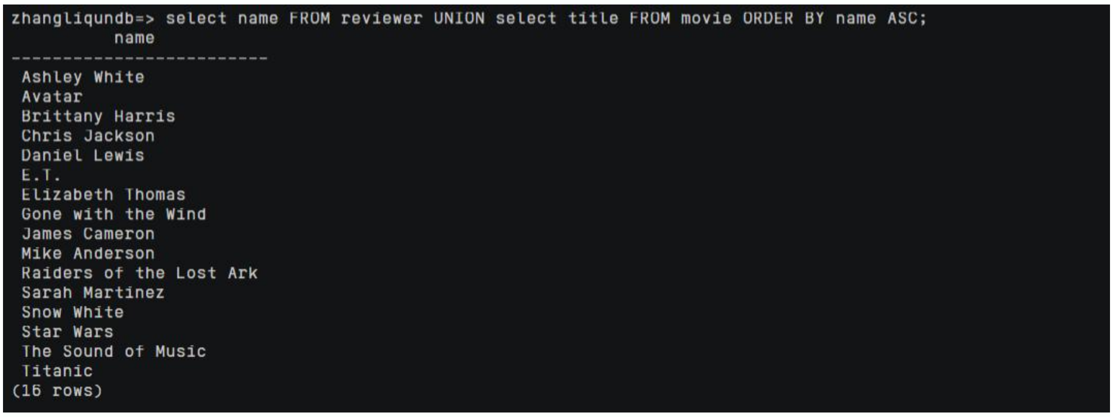
```
zhangliqundb=> SELECT name AS "reviewer name", title AS "movie title", stars FROM movie NATURAL JOIN rating NATURAL JOIN reviewer WHERE director = na
me;
 reviewer name | movie title | stars
---------------+-------------+-------
 James Cameron | Avatar      |     5
(1 row)
```



```
zhangliqundb=> select name "reviewer name", title "movie title", stars from rating natural join movie natural join reviewer G  where director IS NOT
NULL and stars = (select stars from rating natural join movie natural join reviewer where name = director);
 reviewer name | movie title | stars
---------------+-------------+-------
 James Cameron | Avatar      |     5
(1 row)
```

Q16

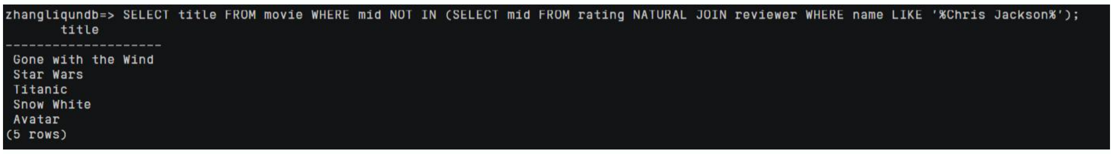
Return all reviewer names and movie names together in a single list, alphabetized. (Sorting by the first name of the reviewer and first word in the title is fine; no need for special processing on last names or removing "The".)



```
zhangliqundb=> select name FROM reviewer UNION select title FROM movie ORDER BY name ASC;
          name
-------------------------
 Ashley White
 Avatar
 Brittany Harris
 Chris Jackson
 Daniel Lewis
 E.T.
 Elizabeth Thomas
 Gone with the Wind
 James Cameron
 Mike Anderson
 Raiders of the Lost Ark
 Sarah Martinez
 Snow White
 Star Wars
 The Sound of Music
 Titanic
(16 rows)
```

Q17

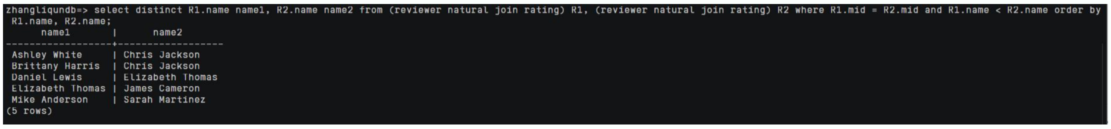
Find the titles of all movies not reviewed by Chris Jackson.



```
zhangliqundb=> SELECT title FROM movie WHERE mid NOT IN (SELECT mid FROM rating NATURAL JOIN reviewer WHERE name LIKE '%Chris Jackson%');
       title
--------------------
 Gone with the Wind
 Star Wars
 Titanic
 Snow White
 Avatar
(5 rows)
```

Q18

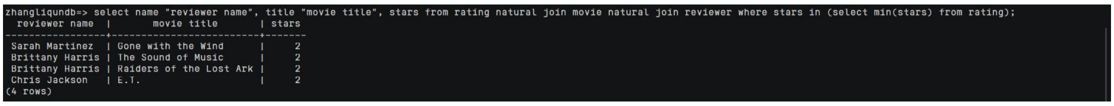
For all pairs of reviewers such that both reviewers gave a rating to the same movie, return the names of both reviewers. Eliminate duplicates, don't pair reviewers with themselves, and include each pair only once. For each pair, return the names in the pair in alphabetical order.



```
zhangliqundb=> select distinct R1.name name1, R2.name name2 from (reviewer natural join rating) R1, (reviewer natural join rating) R2 where R1.mid = R2.mid and R1.name < R2.name order by
 R1.name, R2.name;
      name1       |      name2
------------------+------------------
 Ashley White     | Chris Jackson
 Brittany Harris  | Chris Jackson
 Daniel Lewis     | Elizabeth Thomas
 Elizabeth Thomas | James Cameron
 Mike Anderson    | Sarah Martinez
(5 rows)
```

Q19

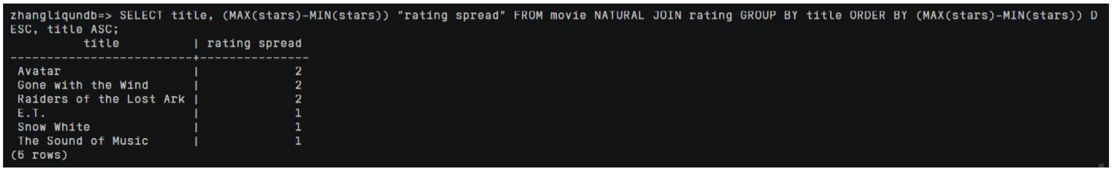
For each rating that is the lowest (fewest stars) currently in the database, return the reviewer name, movie title, and number of stars.



```
zhangliqundb=> select name "reviewer name", title "movie title", stars from rating natural join movie natural join reviewer where stars in (select min(stars) from rating);
  reviewer name  |       movie title       | stars
-----------------+-------------------------+-------
 Sarah Martinez  | Gone with the Wind      |     2
 Brittany Harris | The Sound of Music      |     2
 Brittany Harris | Raiders of the Lost Ark |     2
 Chris Jackson   | E.T.                    |     2
(4 rows)
```

Q20

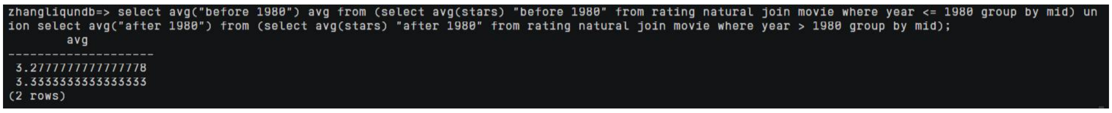
For each movie, return the title and the 'rating spread', that is, the difference between highest and lowest ratings given to that movie. Sort by rating spread from highest to lowest, then by movie title.



```
zhangliqundb=> SELECT title, (MAX(stars)-MIN(stars)) "rating spread" FROM movie NATURAL JOIN rating GROUP BY title ORDER BY (MAX(stars)-MIN(stars)) D
ESC, title ASC;
          title          | rating spread
-------------------------+---------------
 Avatar                  |             2
 Gone with the Wind      |             2
 Raiders of the Lost Ark |             2
 E.T.                    |             1
 Snow White              |             1
 The Sound of Music      |             1
(6 rows)
```

Q21

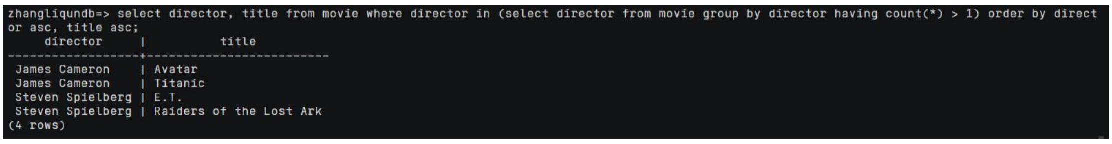
Find the difference between the average rating of movies released before 1980 and the average rating of movies released after 1980. (Make sure to calculate the average rating for each movie, then the average of those averages for movies before 1980 and movies after. Don't just calculate the overall average rating before and after 1980.)



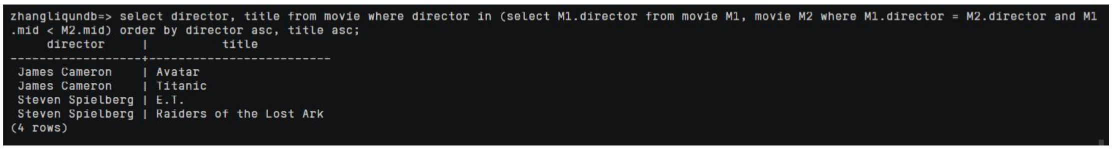
```
zhangliqundb=> select avg("before 1980") avg from (select avg(stars) "before 1980" from rating natural join movie where year <= 1980 group by mid) un
ion select avg("after 1980") from (select avg(stars) "after 1980" from rating natural join movie where year > 1980 group by mid);
        avg
--------------------
 3.2777777777777778
 3.3333333333333333
(2 rows)
```

Q22

Some directors directed more than one movie. For all such directors, return the titles of all movies directed by them, along with the director name. Sort by director name, then movie title. (As an extra challenge, try writing the query both with and without COUNT.)



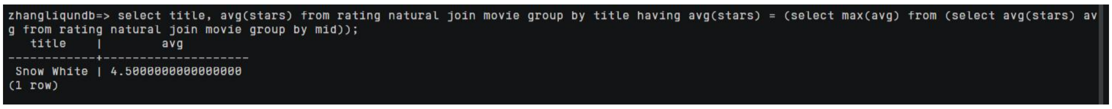
```
zhangliqundb=> select director, title from movie where director in (select director from movie group by director having count(*) > 1) order by direct
or asc, title asc;
     director     |          title
------------------+-------------------------
 James Cameron    | Avatar
 James Cameron    | Titanic
 Steven Spielberg | E.T.
 Steven Spielberg | Raiders of the Lost Ark
(4 rows)
```



```
zhangliqundb=> select director, title from movie where director in (select M1.director from movie M1, movie M2 where M1.director = M2.director and M1
.mid < M2.mid) order by director asc, title asc;
     director     |          title
------------------+-------------------------
 James Cameron    | Avatar
 James Cameron    | Titanic
 Steven Spielberg | E.T.
 Steven Spielberg | Raiders of the Lost Ark
(4 rows)
```

Q23

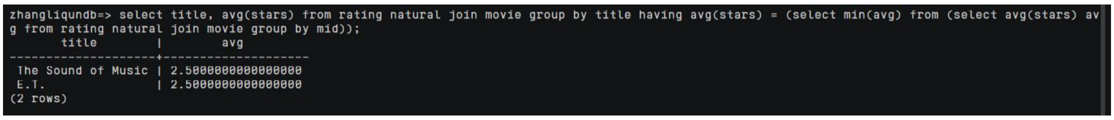
Find the movie(s) with the highest average rating. Return the movie title(s) and average rating. (Hint: This query is more difficult to write in SQLite than other systems; you might think of it as finding the highest average rating and then choosing the movie(s) with that average rating.)



```
zhangliqundb=> select title, avg(stars) from rating natural join movie group by title having avg(stars) = (select max(avg) from (select avg(stars) av
g from rating natural join movie group by mid));
   title    |        avg
------------+--------------------
 Snow White | 4.5000000000000000
(1 row)
```

Q24

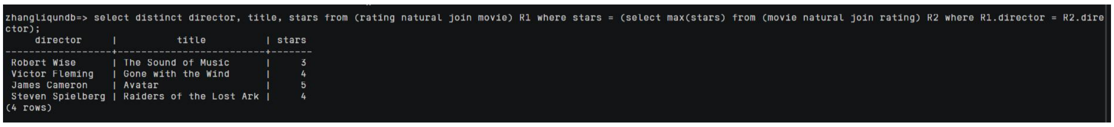
Find the movie(s) with the lowest average rating. Return the movie title(s) and average rating. (Hint: This query may be more difficult to write in SQLite than other systems; you might think of it as finding the lowest average rating and then choosing the movie(s) with that average rating.)



```
zhangliqundb=> select title, avg(stars) from rating natural join movie group by title having avg(stars) = (select min(avg) from (select avg(stars) av
g from rating natural join movie group by mid));
       title       |        avg
-------------------+--------------------
 The Sound of Music | 2.5000000000000000
 E.T.              | 2.5000000000000000
(2 rows)
```

Q25

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.



```
zhangliqundb=> select distinct director, title, stars from (rating natural join movie) R1 where stars = (select max(stars) from (movie natural join rating) R2 where R1.director = R2.dire
ctor);
     director     |          title          | stars
------------------+-------------------------+-------
 Robert Wise      | The Sound of Music      |     3
 Victor Fleming   | Gone with the Wind      |     4
 James Cameron    | Avatar                  |     5
 Steven Spielberg | Raiders of the Lost Ark |     4
(4 rows)
```