SQL results

 $2\,$ $\,$ Find all years that have a movie that received a rating of 4 or 5, and sort them in increasing order.



3 Find the titles of all movies that have no ratings



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 m. title, MAX(r. stars) as highest_rating
FROM Movie m

JOIN Rating r ON m.mID = r.mID
GROUP BY m.mID

ORDER BY m. title
```



For each movie that has at least one rating, find the movie title and total number of stars, the highest star and the person who gave highest star.

```
SELECT t1. title, t1. total_stars, t1. highest_star, re. name as reviewer_name
FROM(

SELECT m.mID, title, SUM(r1. stars) as total_stars, MAX(r1. stars) as highest_star
FROM Movie m

JOIN Rating r1 ON m.mID = r1.mID

GROUP BY title, m.mID

) t1

JOIN Rating r2 ON t1.mID = r2.mID AND t1. highest_star = r2. stars
JOIN Reviewer re ON r2.rID = re.rID

ORDER BY title
```



title	total_stars	highest_star	reviewer_name
Avatar	8	5	James Cameron
E.T.	5	3	Ashley White
Gone with the Wind	9	4	Sarah Martinez
Raiders of the Lost Ark	10	4	Chris Jackson
Raiders of the Lost Ark	10	4	Brittany Harris
Snow White	9	5	Elizabeth Thomas
The Sound of Music	5	3	Chris Jackson

Gone with the Wind

6 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.

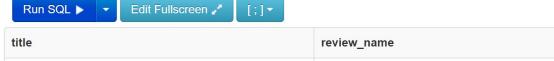
```
SELECT title, name as review_name
FROM Movie m

JOIN Rating r1 ON m.mID = r1.mID

JOIN Rating r2 ON r1.rID = r2.rID

JOIN Reviewer re ON r1.rID = re.rID

WHERE r1.mID = r2.mID AND r1.ratingDate < r2.ratingDate AND r1.stars < r2.stars
```



Sarah Martinez

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.

```
SELECT title, (MAX(stars)-MIN(stars)) AS rating_spread
FROM Movie m

JOIN Rating r ON m. mID = r. mID

GROUP BY m. mID

ORDER BY rating_spread DESC, title
```



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

8 Find the names of reviewers for every director (one row per director with all reviewers).

```
SELECT director, group_concat(distinct name) as reviewers
FROM Movie m

JOIN Rating r ON m.mID = r.mID

JOIN Reviewer re ON r.rID = re.rID

WHERE director IS NOT NULL

GROUP BY director

ORDER BY director
```

