SQL Movie-Rating Query Exercises

COURSE - Databases: Relational Databases and SQL

StanfordOnline

By Njabulo Hlabangana

Background

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

Database

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	<null>
107	Avatar	2009	James Cameron
108	Raiders of the Lost Ark	1981	Steven Spielberg

Reviewer	
rID	name
201	Sarah Martinez
202	Daniel Lewis
203	Brittany Harris
204	Mike Anderson
205	Chris Jackson
206	Elizabeth Thomas
207	James Cameron
208	Ashley White

Rating			
rID	mID	stars	ratingDate
201	101	2	22/01/2011
201	101	4	27/01/2011
202	106	4	<null>
203	103	2	20/01/2011
203	108	4	12/01/2011
203	108	2	30/01/2011
204	101	3	09/01/2011
205	103	3	27/01/2011
205	104	2	22/01/2011
205	108	4	<null>
206	107	3	15/01/2011
206	106	5	19/01/2011
207	107	5	20/01/2011
208	104	3	02/01/2011

Queries

```
--Find the titles of all movies directed by Steven Spielberg.

select title
from Movie
where director = 'Steven Spielberg';
```

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

select distinct year
from movie m join rating r
on m.mID = r.mID
where stars = 4 or stars = 5
order by year
```

```
--Find the titles of all movies that have no ratings.

select title
from movie
where mID not in (select mID from rating)
```

```
--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 join rating using(mID) join reviewer using (rID)
order by name,title,stars;
```

```
--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 distinct name,title
from reviewer join rating using (rID) join movie using(mID)
where rID in
(select r1.rID
from rating r1,rating r2
where r1.rID = r2.rID and r1.mID = r2.mID and r1.ratingDate > r2.ratingDate and
r1.stars > r2.stars)
```

```
--AGGREGATION

--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,stars
from
(select title, max(stars) AS stars
 from movie join rating using(mID)
 where stars >= 1
 group by mID)
 order by title



 --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 spread
from movie join rating using (mID)
group by movie.mID
order by spread desc, title



 --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.)



select round(max(average) - min(average),16)
from
(select avg(average) as average, year
from
(select mID,avg(stars) as average, 'before_1980' as year
from movie join rating using (mID)
where year < 1980 and mID in (select distinct mID from rating)
group by mID
union all
select mID,avg(stars) as average,'after_1980' as year
from movie join rating using(mID)
where year > 1980 and mID in (select distinct mID from rating)
group by mID)
 group by year)
```