**David Jackson**

**Database 2 Assignment 1**

***Queries***

1. *Find the titles of all movies directed by Steven Spielberg.*

**SELECT title FROM movies WHERE director = “Steven Spielberg”;**

1. *Find all years that have a movie that received a rating of 4 or 5.*

**SELECT DISTINCT movie.year FROM movie JOIN rating ON movie.mID = rating.mID WHERE rating.stars = 5 OR rating.stars = 5 ORDER BY movie.year;**

1. *Find the titles of all movies that have no rating*

**SELECT title FROM movie WHERE mID NOT IN (SELECT mID FROM rating);**

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

**SELECT reviewer.name FROM reviewer JOIN rating ON rating.rID = reviewer.rID WHERE rating.ratingDate IS NULL;**

1. *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 the reviewer name, then by movie title and lastly by number of stars.*

**SELECT reviewer.name, movie.title. rating.stars, rating.ratingDate FROM rating JOIN reviewer ON reviewer.rID = rating.rID JOIN movie ON movie.mID = rating.mID ORDER BY reviewer.name, movie.title, rating.stars DESC;**

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

**SELECT movie.title, reviewer.name FROM rating AS r1 JOIN rating AS r2 ON r1.mID = r2.mID JOIN movie ON r1.mID = movie.mID JOIN reviewer ON r1.rID = reviewer.rID WHERE r1.stars > r2.stars AND r1.mID = r2.mID AND r1.rID = r2.rID AND r1.ratingDate > r2.ratingDate;**

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

**SELECT DISTINCT movie.title, rating.stars FROM rating JOIN movie ON rating.mID = movie.mID WHERE rating.stars IN (SELECT max(r1.stars) FROM rating AS r1 WHERE r1.mID = rating.mID) ORDER BY movie.title;**

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

**SELECT movie.title, (SELECT MAX(stars) FROM rating WHERE rating.mID = movie.mID) – (SELECT MIN(stars) FROM rating WHERE rating.mID = movie.mID) AS ratingSpread FROM movie ORDER BY ratingSpread DESC, movie.title;**

1. *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 AVG((SELECT AVG(stars) FROM rating WHERE rating.mID = movie.mID AND movie.year > 1980)) - AVG((SELECT AVG(stars) FROM rating WHERE rating.mID = movie.mID AND movie.year < 1980)) AS yearDiff FROM movie;**

***Modification***

1. *Add the reviewer Roger Ebert to your database, with an rID of 209.*

**INSERT INTO reviewer VALUES (209, ‘Roger Ebert’);**

1. *Insert 5-star ratings by James Cameron for all movies in the database. Leave the review date as NULL.*

**INSERT INTO rating (rID, mID, stars) SELECT ‘209’, mID, ‘5’ FROM movie;**

1. *For all movies that have an average rating of 4 stars or higher, add 25 to the release year. (Update the existing Rows; don’t insert new rows).*

**UPDATE movie SET year = year + 25 WHERE mID In (SELECT mID from rating WHERE stars >= 4);**