SQL Worksheet

Assignment

This worksheet is worth 10% of your course. Submit a single text document (readable either by MS-Word or notepad) named YOURUSERNAME to your IN605 Git repository by September 10th 11:59pm with the statements to do the tasks listed below.

Book Rating Database

You've started a new book-rating website. You've been collecting data on reviewers' ratings of various books. There's not much data yet, but you can still try out some interesting queries. The scripts to create this database in SQLite are on the I: drive. A printout of the data is on the last page

Schema:

Book (bID, title, published, author, genre) Explanation: There is a book with ID number bID, a title, a first published year, a director and a genre.

Reviewer (rID, name) Explanation: The reviewer with ID number rID has a certain name.

Rating (rID, bID, ratings, ratingDate) Explanation: The reviewer rID gave the book bID a number of ratings (1-5) on a certain ratingDate.

Tasks

Create SQL statements to achieve the following:

Use the techniques taught in class - zero marks for using a NATURAL JOIN or USING instead of ON

Queries

1. Find the titles and the genre of all books written by Harper Lee.

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

3. Find the names of all reviewers who rated To Kill a Mocking Bird.

4. 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.

5. For any rating where the reviewer is the same as the author of any book, return the reviewer name, book title, and the ratings

6. Write a query to return the rating data in a more readable format (usings titles): reviewer name, book title, ratings, and ratingDate. Also, sort the data, first by reviewer name, then by book title, and lastly by ratings.

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

8. For each book that has at least one rating, find the highest rating that book received. Return the book title and the rating. Sort by book title.

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

10. Find the difference between the average rating of books released before 1970 and the average rating of books released after 1970. (Make sure to calculate the average rating for each book, then the average of those averages for books before 1970 and books after. Don't just calculate the overall average rating before and after 1970.)

Modification

11. Add the reviewer John Green to your database, with an rID of 209.

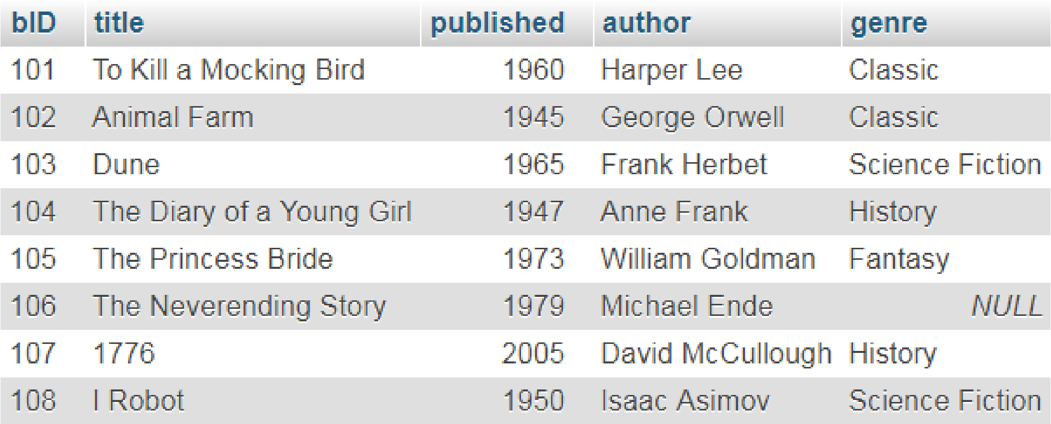
12. Insert 5-star ratings by Daniel Lewis for all books in the database. Leave the review date as NULL. Do Not write five insert statements.

13. For all books that have an average rating of 4 or higher, add 25 to the published year. (Update the existing Rows; don't insert new Rows.)

14. Remove all ratings where the book year is before 1970 or after 2000, and the rating is fewer than 4.

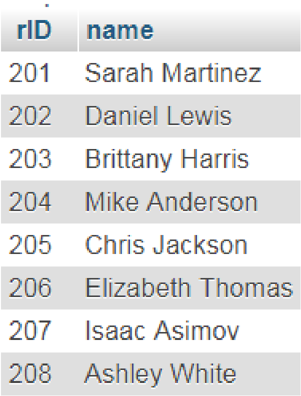
Data

Book



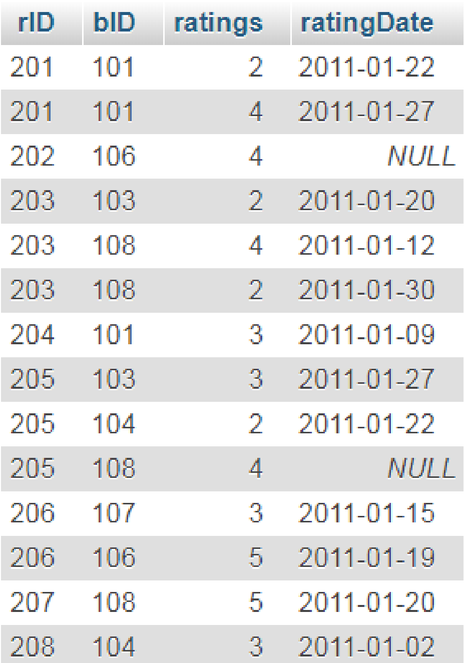
bID	title	published	author	genre
101	To Kill a Mocking Bird	1960	Harper Lee	Classic
102	Animal Farm	1945	George Orwell	Classic
103	Dune	1965	Frank Herbet	Science Fiction
104	The Diary of a Young Girl	1947	Anne Frank	History
105	The Princess Bride	1973	William Goldman	Fantasy
106	The Neverending Story	1979	Michael Ende	*NULL*
107	1776	2005	David McCullough	History
108	I Robot	1950	Isaac Asimov	Science Fiction

Reviewer



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

Rating



rID	bID	ratings	ratingDate
201	101	2	2011-01-22
201	101	4	2011-01-27
202	106	4	*NULL*
203	103	2	2011-01-20
203	108	4	2011-01-12
203	108	2	2011-01-30
204	101	3	2011-01-09
205	103	3	2011-01-27
205	104	2	2011-01-22
205	108	4	*NULL*
206	107	3	2011-01-15
206	106	5	2011-01-19
207	108	5	2011-01-20
208	104	3	2011-01-02