

RAT 1 Homework

Who needs to do this homework?

for those who earned less than 25 on the first RAT this hw is mandatory.

You need to get at least 90% of the questions correct. You cannot take the next RAT until you complete this work.

optional for everyone else (Completing Part 3 can gain you 10xp)

You need to get at least 90% of the questions correct.

Alternatively,

You can gain 5 points by doing question 10.

Instructions

I want you to answer these questions by writing SQL statements into a file that I can load into Postgresql. Your grade will be a combination of a visual inspection of the file and running the file in postgres.

the file you submit should run in postgres.

Here are sample questions

sample1. Given the information below write SQL instructions to create the *movie* table.

Attribute Name	Description
movie	The title of the movie. a maximum of 35 characters
boxoffice	The box office amount
tmeter	The Rotten Tomato rating expressed as in integer
rating	The ratings (e.g., PG, R)
mlength	the length of the movie in minutes (an integer)
genre	A string of up to 35 characters representing the genres
released	the date the movie was released

sample2. Write SQL statements to populate the *movie* table with the data shown here:

movie	boxoffice	tmeter	rating	mlength	genre	released
Mr & Mrs Smith		63	unrated	95	comedy	1941-01-09
The Wedding Ringer	39.4	33	R	101	comedy	2015-01-16

sample 3. Write a query to show the movie title and tmeter rating for all the movies in the database.

To answer these sample questions you would write an sql file containing the following:

```
-- CS350 HW 1

-- Question 1
CREATE DATABASE hw1;
\c hw1
CREATE TABLE movies (movie varchar(35),
                      boxoffice dec(4,1),
                      tmeter integer,
                      rating varchar(6),
                      mlength integer,
                      genre varchar(35),
                      released date);

-- Question 2
insert into movies values ('Paddington', 39.9, 98, 'PG', 95, 'comedy', '2015-01-09');
insert into movies values ('The Wedding Ringer', 39.4, 33, 'R', 101, 'comedy',
'2015-01-16');

-- Question 3
select movie, tmeter from movies;
```

you will attach the file to an email (subject 'hw1') to
submit.o.bot@gmail.com

ANY SYNTAX ERRORS WILL RESULT IN A ZERO FOR THAT
QUESTION

section 1

1. Given the information below write SQL instructions to create the *programmer* table.

Attribute Name	Description
EmpNo	unique id, max is three characters, This is a required field
Project	Project in which programmer participates. Max is 3 characters. The default is NEW.
TaskNo	Number (integer) of the task associated with the project.
Last_Name	Surname of employee. max is 25 characters. This is a required Field
First_Name	Employees's first name. Max is 25 characters
Hire_Date	Date when the employee was hired
Language	Programming Language used by employee. Max is 15 characters
Clearance	Type of clearance given to employee.

Here is a sample table:

EmpNo	Last_Name	First_Name	Hire_Date	Project	Language	Task	Clearance
201	Campbell	Jeanne	1/1/2013	XYZ	Cobol	52	Ultra Secret
152	Davies	Stephen	8/3/2001	KOO	Fortran	13	Secret
253	Finlayson	Ian	8/12/2012	SYL	Forth	29	Top Secret
192	Marshall	Andrew	8/03/2014	DAT	Algol	93	Top Secret

390	Anewalt	Karen	5/21/1999	SCR	Pascal	13	Secret
111	Mulkern	Ann	5/4/1998	XYZ	Forth	12	Ultra Secret
123	Scully	Dana	7/30/2000	XFI	Forth	13	Ultra Secret

2. Write SQL statements to populate the *programmer* table with the data shown above.

3. Write SQL queries for the following:

- display first name, last name, clearance, and hire date (in that order) for all employees.
- display first name, last name, project and language for all employees who know Forth.
- display first name, last name, project and language for all employees who know Cobol or Fortran
- display first name, last name, clearance and language for all employees who know Forth and have Ultra Secret Clearance.
- Display all information for Stephen Davies.

4. Write the SQL query to show the structure of the *programmer* table.

section 2

5. Write the SQL statements to create the following *books* table

Attribute Name	Description
ISBN	unique 13 digit number
Title	Title of the book
Author_Last_Name	author's last name (assuming book only has one author)
Author_First_Name	first name
Pub_Date	publication year
Publisher	The Publisher
Pages	number of pages

6. Write SQL statements to populate the *books* table with the following data:

Title	Author_First_Name	Author_Last_Name	Pub_Date	Publisher	Pages	ISBN
Sit Down and Shut Up	Brad	Warner	2015	Wisdom	232	9781614293163
Method of No-Method	Sheng	Yen	2008	Shambhala	144	9781590305751
Shattering the Great Doubt	Sheng	Yen	2009	Shambhala	208	9781590306215
Compass of Zen	Seung	Sahn	NULL	Shambhala	394	9781570623295
Art of Just Sitting	John	Loori	2002	Wisdom	256	9780861713943
Essential Dogen	Kazuaki	Tanahashi	2013	Shambhala	224	9788129118332
Shobogenzo	Gudo	Nishijima	2006	BookSurge	334	9781419638206
Head First SQL	Lynn	O'Beighley	NULL	O'Reilly	610	9780596526849

7. Write SQL queries for the following:

- Show me the title of books whose titles start with 'S'
- Show me the title author's first and last names, and the number of pages for books between 200 and 300 pages inclusive.
- Show me all the books (titles, and publisher) not published by Shambhala.
- Show me all book titles that contain the letter 'o'.
- Show me the titles of books written by Lynn O'Beighley
- show me all books (titles, author's first and last names) which do not have a publication date (that is, it is NULL)
- Show me all the books (titles, and publisher) not published by Shambhala, Wisdom, or O'Reilly.

section 3.

8. In the sample question, I gave the structure of the movie table. Please redo the CREATE TABLE to

- make the movie name default to "" (the empty string) if one is not provided
- make the genre a required field (it cannot be null)

9. Write SQL statements to add 12 movies to the movies database (you can use the data from the first RAT, or make your own). Please have some release dates be NULL.

10. Write SQL queries for the following ([using the table from RAT #1](#))

- a. show me the movie name and the rating for science fiction movies. For example, in the RAT 1 data Guardians of the Galaxy was listed as action/science fiction/fantasy, and I would like that movie (among others) returned.
- b. show me movie names, genres, and tmeter ratings of movies with a tmeter rating over 80.
- c. Show me movies (titles and release dates) that do not have a NULL release date.
- d. Show me movies (titles only) whose titles start with a vowel.