CS 437/537 (Fall 2017)

Assignment 2

Published: Sep. 20, 2017
Due: Sep. 30, 2017 (11:59pm)

If you know you are going to submit your assignment late, please let us know in advance (send an email to cs437ta@cs.yale.edu). Any and all resources may be used as long as you cite them, with the exception of collaborating with other people.

*** You can download Postgress here:**
https://www.postgresql.org/download/

*** You can find some SQL tutorials on the postgres website:**
https://www.postgresql.org/docs/8.4/static/tutorial-concepts.html
https://www.postgresql.org/docs/8.4/static/tutorial-advanced.html

If you have ANY questions, please do not hesitate to let us know (email, office hours, etc.)

Important requirements: Please do not submit test data, results, etc. – just submit the SQL queries you wrote (i.e., .sql file or .txt file). Also, attribute names do not have spaces (which might be difficult to see when underlined) – for example, "employee_name" is actually "employee_name".

Important – Make sure to test your SQL code, and create sample databases to test it.

Part 1: **Bookstore** (same as in Homework 1): (15pts)
The underlines represent the primary keys for each table. "Store" is a foreign key referencing the bookstore id. And "author" (in book) is a foreign key referencing the author id in the author table.

Bookstore(id, name, city, address)
Book(book id, title ,author ,price ,store)
Author(author id, name, city, address, age)

Write **SQL queries** to find:

1. Find all bookstore addresses and names in the city New Haven. (3)
2. Find all author names who have at least one book under 10 dollars. (3)
3. Find all the titles of books where the author is over 40. (3)
4. Find all names and addresses of authors who live in a city which has a bookstore selling their books. (3)
5. Find out the address, name, and city of the bookstore who is selling the cheapest book called "Game of thrones". (3)

Part 2: META Equity Trading Associates (50 points)

META Equity Trading Associates are a (hypothetical) new firm on Wall Street. This is its employee database schema (primary keys are underlined):

employee(employee_name, street, city)
works(employee_name, department_name, job_title, salary)
department(department_name, city)
manages(employee_name, manager_name)

Write SQL queries to find:

1. Names of employees who work in the 'Income Tax Restructuring' department. (5)
2. Name, salary and address (street, city) of META's CTO. The IRS is extremely interested in the results of this query. (5)
3. Names of employees who earn more than the average salary. (5)

4. Names of employees who live in cities other than their department's headquarter city. (5)
5. Names of employees who live in the same cities and on the same streets as their managers. (7)
6. Names of employees who earn more than the average salary of their department. (8)
7. Department with largest payroll (largest sum of all employee salaries). (7)
8. Department with largest payroll per head. (8)

Part 3: Easy Vehicle Insurance LLC (20 points)

Easy Vehicle Insurance LLC keeps track of accident reports in a database with the following schema (primary keys are underlined):

person(license_number, name, address)
car(car_regnum, model, year)
accident(report_number, date, location)
owns(license_number, car_regnum)
participated(license_number, car_regnum, report_number, damage_amount)

Write SQL queries to find:

1. Number of accidents in which cars belonging to "John Smith" were involved (7)
2. Average damage amount for each car model for all accidents that occurred since Jan 1, 2016 sorted from highest to lowest. (7)
3. Drivers who were not involved in any accidents since Jan 1, 2016. (7)

Part 4: Fortunoff Library Internet Reservation Terminal (15 points)

The Fortunoff Library Internet Reservation Terminal system keeps track of books, and reports the results to the librarians. Their database has the following schema (as usual, primary keys):

student(student_id, name)
books(isbn, title, author, publisher)
loan(student_id, isbn, issue_date, due_date)

Write SQL queries to find:

1. For each author, names of students who borrowed more than three books by that author. (5)
2. Names of students who borrowed all books written by 'Avi Silberschatz'. (5)
3. Names of students who borrowed books written by anyone with the last name "Sade" (assume that names are stored as 'first middle last'). (5)