CS127 Homework 3

Due: October 8th, 2014 3:00PM

Warmup 1 (Textbook Problem 3.5)

Suppose that we have a relation $marks(ID, score)$ and we wish to assign grades to students based on the score as follows: grade F if $score < 40$, grade C if $40 \leq score < 60$, grade B if $60 \leq score < 80$, and grade A if $80 \leq score$. Write SQL queries to do the following:

a. Display the grade for each student, based on the marks relation.

b. Find the number of students with each grade.

Warmup 2 (Textbook Problem 3.17)

Consider the relational database of Figure 3.20 in the textbook. Give an expression in SQL for each of the following queries:

a. Give all employees of "First Bank Corporation" a 10 percent raise.

b. Give all managers of "First Bank Corporation" a 10 percent raise.

c. Delete all tuples in the *works* relation for employees of "Small Bank Corporation".

Warmup 3 (Textbook Problem 3.21)

Consider the library database of Figure 3.21 in the textbook. Write the following queries in SQL.

a. Print the names of members who have borrowed any book published by "McGraw-Hill".

b. Print the names of members who have borrowed all books published by "McGraw-Hill".

c. For each publisher, print the names of members who have borrowed more than five books of that publisher.

d. Print the average number of books borrowed per member. Take into account that if a member does not borrow any books, then that member does not appear in the *borrowed* relation at all.

Problem 4 (To Be Graded)

Consider again the simplified university registrar database from the previous homeworks:

Student		
name	**gradyear**	**gpa**
Amy	2016	3.95
Ben	2015	3.87
Carl	2016	3.29
Dan	2017	3.43
Eliza	2015	4.0

Course		
title	**semester**	**instructor**
CS33	2014F	Doeppner
CS127	2014F	Zdonik
CS195	2013F	Kraska
CS127	2012F	Zdonik
CS136	2012S	Fonseca

Enrollment			
name	**title**	**semester**	**grade**
Eliza	CS33	2014F	A
Eliza	CS127	2014F	A
Ben	CS127	2012F	A
Carl	CS195	2013F	C
Carl	CS127	2014F	B

The keys for each relation are as follows:

- *Student:* name (all student names are assumed to be unique)
- *Course:* title and semester
- *Enrollment:* name, title, and semester

For each of the following, give the equivalent SQL query:

1. Find all students graduating in 2015 and their grade in CS127, if they have ever taken it.
2. Find all students with a GPA greater than the average GPA for their grad year.
3. Find all of Prof. Doeppner's courses from this semester (2014F) with at least one student graduating in 2015 enrolled.
4. Find all instructors who are teaching at most one course this semester.
5. Find all instructors with at least one course this semester whose enrollment count is less than the average number of enrollments for all courses this semester.
6. **Extra Credit:** Find all distinct pairings of students enrolled in CS127 this semester, where students cannot be paired with themselves (e.g., (Amy, Amy)) and inverted pairings are considered equivalent (e.g., $(Amy, Ben) \equiv (Ben, Amy)$).