# CS 437 / 537: Database Systems

# Assignment 2: Modules 3 and 4

Assigned: Sunday 02 February, 2014 **Due: 2359hrs Tuesday 18 February, 2014**


**Instructor:** Avi Silberschatz, AKW 214
**TAs:** Debayan Gupta, Dongqu Chen
**Course page:** http://zoo.cs.yale.edu/classes/cs437
**Email:** cs437.s14@gmail.com

**Total points:** 70

## Instructions

- All HWs should be emailed to cs437.s14@gmail.com. You should have two copies of your code. One in a primary solution file in any readable format like PDF with the rest of your answers, and the other in a separate .txt or .sql file.
- Make sure you have tested your code before submission. Run:

```
psql -h lab.zoo.cs.yale.edu -U username -f yourcodefile
```

- Your subject line should be "**Assignment 2**". If you resubmit, it should be "**Assignment 2 Resubmission**". Please put your name in the filenames of all files you submit. E.g., Debayan's submission might be "Debayan-Gupta-a2.pdf" and "Debayan-Gupta-a2-code.sql".
- For the coding portion, you'll need to create the tables, insert some dummy data, and then test your queries. Please make sure you're not missing semicolons, etc. when you submit. We really don't want to remove points for such pointless reasons.
- You are free to use any and all resources you can access, as long as you do the work yourself, and cite all sources. This means that you can find your definitions on Wikipedia, search for solutions to SQL issues on forums, etc. You should know this already, but we'll say it anyway: You may not collaborate with other students or get other people to do your HW for you.
- Check your attachments carefully before you click send.

## Part I: Reading *(20 points)*

As part of each assignment, you are expected to read and critique a paper. Your critique should be in 2 parts, each ~5-10 lines:

1. Summary of the paper. Highlight the primary technical innovations that the paper talks about.
2. Errors, improvements, whatever you found interesting about the paper.

For this HW, your paper is MapReduce (http://research.google.com/archive/mapreduce-osdi04.pdf). This isn't a pure DB paper, but it's really important in the world of big data.

## Part II: Entity-Relationship Diagram *(15 points)*

You'll learn more about this in class over the coming week, so hold off working on this question for the next few days – remember, you've got over 2 weeks!

There are two types of students at Yale - undergrads and grads. Undergrads take one or more courses. Grads can only take courses that are at a graduate level (they can also grade undergrads). Professors can teach undergrads and advise grads.

- Undergrads live in colleges (it doesn't matter where grads live, rumor has it they live in their offices). (2)
- An undergrad takes one or more courses. (2)
- Undergrads pay, but grads get paid. (2)
- A grad takes zero or more courses but at a graduate level. (3)
- Grads grade zero or more courses. (2)
- A Professor teaches zero or more courses. (2)
- Professors advise zero or more Super-Grads. (2)
- *Extra credit:* A grad cannot grade a course unless his/her advising Professor is also teaching that course. (5)

Model **only** the above information using an E-R diagram. If you can interpret the above in multiple ways, state your assumptions and continue. Don't worry too much – we're not out to slash your score. We **want** to give you a high grade. Just write down whatever you think is right.

## Part III: Coding *(35 points)*

### 1. META Equity Trading Associates **(15 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. (1)
2. Name, salary and address (street, city) of META's CFO. The IRS is extremely interested in the results of this query. (1)
3. Names of employees who earn more than the average salary. (1)
4. Names of employees who live in cities other than their department's headquarter city. (2)
5. Names of employees who live in the same cities and on the same streets as their managers. (3)
6. Names of employees who earn more than the average salary of their department. (2)
7. Department with largest payroll (largest sum of all employee salaries). (3)
8. Department with largest payroll per head. (2)

## 2. Easy Vehicle Insurance LLC (10 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 'Debayan Gupta' were involved (I *really* need to know this). (1)
2. Average damage amount for each car model for all accidents that occurred since Jan 1, 2014 sorted from highest to lowest. (3)
3. Drivers who were not involved in any accidents since Jan 1, 2014. (2)
4. Number of accidents per car model since Jan 1, 2014 (show models with 0 accidents). (4)

## 3. Fortunoff Library Internet Reservation Terminal (10 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. (3)
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'). (2)