

**Written Assignment #1**  
**CS 6530 Fall 2017**  
**Due: Sep 12.**

**Problem 1.** [20pts]

Consider a database with the following schema (when more than one attribute is underlined, the combination of them is the primary key):

Customer (cname, address)  
Dealership (dname, address)  
Car (model, manufacturer)  
Visits (cname, dname, times\_a\_year)  
Likes (cname, model)  
Serves (dname, model, price)

Write the following queries in **relational algebra**:

- (1) Find all customers who has visited “Ford Provo” (“Ford Provo is a dealership name).
- (2) Find all dealerships that serve both Toyota and Ford.
- (3) Find all dealerships that serve at least one of the car models John likes for no more than \$10,000.
- (4) For each dealership, find all car models served at this dealership that are liked by none of the customers who visit that dealership.
- (5) Find all customers who visit *only* those dealerships that serve some car models they like.
- (6) Find all customers who have visited *every* dealership that serves some car models they like.
- (7) Find those customers who like all cars Mike likes.
- (8) Find those customers who like exactly the same set of cars as Mike does.
- (9) For each car model, find the dealership(s) that serves it at the lowest price.
- (10) Find the dealership name(s) visited by all customers from SLC (SLC is an address).

**Problem 2.** [32pts]

Consider the following schemas, that are available from our database server in the acmdb database (acmdb).

student(sid, sname, sex, age, year, gpa)  
dept(dname, numphds)  
prof(pname, dname)  
course(cno, dname, cname)  
major(dname, sid)  
section(dname, cno, sectno, pname)  
enroll(sid, dname, cno, sectno, grade)

We showed how to connect to the database server in class. You can use either MySQL Workbench or connect via linux/unix mysql client by “mysql -u cs6530user -p -h georgia.eng.utah.edu cs6530db”. (password was announced in class). Once connected, you can “show tables”, “describe tablename”, and type any SQL statements (must end with ;) etc. (you might need to ssh to georgia.eng.utah.edu first)

Answer the followings using SQL queries. Submit **both your SQL queries and the query results from the database server**. Note that you can store all your SQL queries in a txt file (e.g., query.txt, end each SQL with a semicolon and separate each SQL query with an empty line), and execute them all in once by “source query.txt” (if you are using the mysql client to connect to the DB server). An example of query.txt with 3 queries is given below:

Select \* from student;

```
Select dname
from
dept;
```

```
Select * from major;
```

To capture the query results from the database server in Unix/Linux, you can use the “script” command. In particular, when you are ready to execute all your queries from query.txt. Do the followings:

- 1) script output.txt
- 2) connect to the DB server and “source query.txt;”
- 3) quit the server by “quit” or “exit”
- 4) type ctrl+d to end the scripting process.
- 5) all screen printout will be captured in output.txt.

### Questions:

1. What is the age of the oldest student.
2. Find the names and gpas of the students who have enrolled in course 302.
3. Find the names and majors of students who have taken an advanced course (i.e., the course title contains a keyword ”Advanced” somewhere).
4. Find the names of students who have enrolled in both a course offered by the “Computer Sciences” department and a course offered by the “Mathematics” department.
5. For each department, find the average age of the students majoring in that department along with the age difference between the oldest and youngest students.
6. How many students have more than one major?
7. Find the name(s) of the student(s) who have taken the most number of courses (same cno with different sectno will be viewed as the same course for this purpose).
8. Find the name(s) of the youngest 2nd year student(s) (i.e., year = 2).
9. Print the ids, names, and gpas of the students who have taken all Computer Sciences courses.
10. For those departments that have no majors (i.e., students who major in that department) taking a “Computer Sciences” course, print the department name and the number of PhD students in the department.
11. Find the student names for each age group with the maximum gpa.
12. Find the name(s) of the professor(s) who has (have) taught the most number of courses (multiple sections of the same course should be counted as multiple teaching assignments for this purpose).

### Problem 3. Relational Algebra [16pts]

Consider the dvdrental schema from the dvdrental database on our database server: <http://www.postgresqtutorial.com/postgresql-sample-database/>. Answer the following queries in relational algebra:

1. Find the customer ids who have rented all films.
2. Print the store id and the phone number for each store that has only one staff member.
3. Find every pair of customer and staff who are from the same city, by listing their customer\_id and staff\_id.
4. Find the customer ids who *have rented all and only those* films acted by Emily Dee.

### Problem 4. [32pts]

We showed how to connect to the database server in class (and in the class website). You can use any PostgreSQL client program (e.g., pgadmin) or connect via linux/unix from one of the CADE machines or shell.cs.utah.edu by “psql -h georgia.eng.utah.edu -U cs6530user dvdrental”. (password has been posted on the announcement section in canvass). Once connected, you can “\t”, “\dt”, “\d”, “\d+”, “\d+ tablename” etc.

Answer the followings using SQL queries. Submit **both your SQL queries and the query results from the database server**. Note that you can store all your SQL queries in a txt file (e.g., query.txt, end each SQL with a semicolon and separate each SQL query with an empty line), and execute them all in once by “\i query.txt”. An example of query.txt with 3 queries is given below:

```
Select * from country;
```

```
Select count(*)  
from  
actor;
```

```
Select count(*) from city;
```

To capture the query results from the database server in Unix/Linux, you can use the “script” command. In particular, when you are ready to execute all your queries from query.txt. Do the followings:

- 1) script output.txt
- 2) connect to the DB server and “\i query.txt;”
- 3) quit the server by “quit” or “exit”
- 4) type ctrl+d to end the scripting process.
- 5) all screen printout will be captured in output.txt.

### Questions:

1. Find the customer ids who have rented all films.
2. Print the store id and the address (only need to show the “address” field) for each store that has only one staff member.
3. Find every pair of customer and staff who are from the same city, by listing their customer\_id, staff\_id and city name.
4. Find the customer ids who *have rented all and only those* films acted by Emily Dee.
5. Find the address and the total inventory count for the store that has the largest inventory (judged by the count over inventory id).
6. Find the first name, last name, and the number of films acted by the actor who has acted in the least number of films.
7. Print the city name, the total amount of payments by customers from that city, and the total number of customers from that city for each city. Sort the output by the total payment (high to low) and break ties by total number of customers (high to low) and limit the output to top 15 cities.
8. For each country that starts with ‘A’, show the country name and total number of customers in that country; sort the output by the customer count from high to low.