

## CS352 - DATABASE SYSTEMS

### Introduction to SQL Homework Problems

As noted in the syllabus, most of the homework sets in the course will include some problems designed to help you develop facility with Structured Query Language (SQL). Early assignments will largely be a review of concepts you have already seen in CS211, while later assignments will involve you with new features of SQL, more or less correlated to the theoretical topics being discussed in class.

Many of the problems will involve a database called `sample` that is distributed with DB2. The tables in this database correspond to the attached ER diagram. The commands used to create this database and the complete contents of each table are listed in the Appendix of the Birchall book. *Spend some time familiarizing yourself with the structure of this database before attempting the homework problems that pertain to it!*

You will be expected to dig out the information needed to answer these problems from the following sources. Since they are .pdf files, you will probably want to download them onto your computer for easy access. (Some suggestions as to where to look may be included in the homework) Generally speaking, you will find the Birchall book to be the best place to start; however, he does not cover all of the features of SQL.) They are accessible from the blackboard site for the course.

Birchall, Graeme. *DB2 UDB V8.2 Cookbook*.

IBM. *IBM DB2 Universal Database SQL Reference* (2 volumes)

For these problems, you will be expected to turn in a printout of the query (neatly formatted using multiple lines and indentation) and the results generated by executing the query, created by selecting an appropriate portion of the screen and printing it out. For example, if the question were “what is all the information stored in the org table”, the following printout would suffice. *Manually verify the output printed by db2 against the table contents in Birchall to be sure you’ve gotten the right answer - this becomes increasingly important as the assigned queries become more complex!*

```
select *
  from org;
```

DEPTNUMB	DEPTNAME	MANAGER	DIVISION	LOCATION
10	Head Office	160	Corporate	New York
15	New England	50	Eastern	Boston
20	Mid Atlantic	10	Eastern	Washington
38	South Atlantic	30	Eastern	Atlanta
42	Great Lakes	100	Midwest	Chicago
51	Plains	140	Midwest	Dallas
66	Pacific	270	Western	San Francisco
84	Mountain	290	Western	Denver

8 record(s) selected.

### Abbreviated Usernames for DB2

DB2 limits user id's to 8 characters, and in one case only 7 characters are used. Since your linux username is usually of the form *firstname.lastname*, you will be given a special abbreviated alias for work with DB2. This alias will access the same linux account as your full username (and you can use either when logging in to linux), but you must use the abbreviated alias whenever you interact with DB2.

## General Procedures for SQL Homework Problems

1. Log in to one of the linux workstations. These have a client version of db2 installed on them. (For this purpose you can use either your regular or abbreviated username).
2. Issue the following command. (Note that the first character on the line is a dot, and one space separates it from the rest of the line. Also note that db2 limits certain names to 8 characters - hence "clien" not "client"!)  

```
. ~db2clien/sqlllib/db2profile
```

(This serves to define various commands and symbols needed by db2, using a "profile" associated with the client installation of db2. Alternately, you can put the above line into the .bashrc file in your home directory, and this task will be done automatically whenever you login to a workstation.)
3. Issue the following command to start db2 with semicolon as the statement terminator and no prompting:  

```
db2 -t +p
```
4. Issue the following command to db2 to connect to the sample database:  

```
connect to sample user your-abbreviated-username ;
```

then type your Linux password when prompted to do so. (This serves to authenticate you to the db2 server where the database resides).
5. Issue the following command to db2 to avoid having to type the schema name in front of the name of each table:  

```
set schema db2inst1;
```
6. You can edit the previous command using your favorite editor by using a command like  

```
edit editor pico;
```

(Note: db2 loses formatting information when editing a command, so you may have to re-edit and select and print in pico to produce well-formatted output.)
7. When you are all through with the db2 session, issue the following command to db2:  

```
terminate;
```

### Structure of the sample Database

As you can see from the diagrams on the following page, the *sample* database consists of two groups of tables containing somewhat similar information, plus several independent tables. This is not meant to be an example of good database design, but rather as a basis for SQL examples. (Most of the examples in the Birchall book use this database, as does some of the documentation published by IBM).

One group of tables (*org* and *staff*) records information about people working for various departments within a company. The "works for" relationship is represented by each entry in the *staff* table containing a foreign key (*dept*) referring to the primary key (*deptnumb*) of the appropriate row in the *org* table. The "manager" relationship is represented by each entry in the *org* table containing a foreign key (*manager*) referring to the primary key (*id*) of the appropriate row in the *staff* table.

Another group of tables has a similar structure involving the *department* and *employee* tables (though the field names are different). In addition, this group of tables includes one that records information about projects (*project*) - each of which is associated with a specific department, and is the responsibility of a particular employee. Moreover, a project may be a part of a larger project. (This information is represented by the foreign keys *deptno*, *respemp* and *majproj*). For each

project, the `emp_act` table records information about various activities - each performed by a particular employee. Note that this must be an entity, not a relationship, because a given employee may be involved in several activities for a given project. Finally, for some of the employees, their photo and/or resume is stored in the database as a binary large object (`blob`) or character large object (`blob`).

There are also three standalone tables not connected to any other tables - in fact, two don't initially contain any data. (They are for practice with `insert`).

