**Lab 4.**

Go to <http://users.csc.calpoly.edu/~lstanche/csc365/ra/> and download both files in a new directory. Check <https://www.cs.duke.edu/~junyang/ra/> for more information. Update the sample.properties file and put your database name, login, and password. Execute the program by running

java –jar ra.jar sample.properties

Here is an example of how to use the program.

ra> \select\_{true} Students;

Output schema: (id INT, name VARCHAR)

-----

2|343

3|353

-----

Total number of rows: 2

Here is an overview of the supported operations.

* \select\_{*cond*} is the relational selection operator. For example, to select Drinker tuples with name Amy or Ben, we can write \select\_{name = 'Amy' or name = 'Ben'} Drinker;. Note that string literals should be enclosed in ***single*** quotes, and you may use boolean operators and, or, and not. Comparison operators <=, <, =, >, >=, and <> work on both string and numeric types.
* \project\_{*attr\_list*} is the relational projection operator, where *attr\_list* is a comma-separated list of attribute names. For example, to find out what beers are served by Talk of the Town (but without the price information), we can write \project\_{bar, beer} (\select\_{bar = 'Talk of the Town'} Serves);.
* \join\_{*cond*} is the relational theta-join operator. For example, to join Drinker(name, address) and Frequents(drinker, bar, times\_a\_week) relations together using drinker name, we can write Drinker \join\_{name = drinker} Frequents;. Tables must have different attribute names. If they do not, then use the rename operator.
* \cross is the relational cross product operator. For example, to compute the cross product of Drinker and Frequents, we can write Drinker \cross Frequents;.
* \union is the relational union (set version). For a trivial example, to compute the union between Drinker and itself, we can write Drinker \union Drinker.
* \rename\_{*new\_attr\_name\_list*} is the relational rename operator, where *new\_attr\_name\_list* is a comma-separated list of new names, one for each attribute of the input relation. For example, to rename the attributes of relation Drinker and compute the cross product of Drinker and itself, we can write \rename\_{name1, address1} Drinker \cross \rename\_{name2, address2} Drinker;.

Write the following queries in relational algebra. Populate the tables so that the queries give meaningful output.

1. Print the SSN of John Smith.

\project\_{SSN} (\select\_{Name = 'John Smith'}Customer);

1. Print the numbers of all the credit cards of John Smith.

\project\_{CCNum} (Ownership \join\_{CustomerId = Id} (\select\_{Name = 'John Smith'} Customer));

1. Print all the transaction information from January, 2015 for credit card number 1236666.

\select\_{Date > '2015/1/1' and Date < '2015/1/31' and CCNum = '1236666'} Transaction;

1. Print the credit limit for all credit cards of Maria Johnson.

\project\_{cclimit}(CreditCard \join\_{Number = CCNum} (\project\_{CCNum} (Ownership \join\_{CustomerId = Id} (\select\_{name = 'Maria Johnson'} Customer))));

1. Print the names of vendors who have transaction on January 2nd, 2015.

\project\_{name}(Vender \join\_{Id = vid}\project\_{vid}(\select\_{Date = '2015/1/2'} Transaction));

If your last name is A-G, e-mail Vivian Fong [vfong01@calpoly.edu](mailto:vfong01@calpoly.edu)  
If your last name is H-P, e-mail Esha Joshi <[ejoshi@calpoly.edu](mailto:ejoshi@calpoly.edu" \t "_blank)>  
If you last name is Q-Z, e-mail Derek Chan <dchan17@calpoly.edu>

Due date: October 29th, 2015