

CS 122A: Introduction to Data Management – Spring 2016, UC Irvine

Prof. Chen Li

Homework 3: Relational Algebra and SQL query (100 points)

Due Date: Thursday, April 28, 2016 11:45 PM, on EEE

Submission

Please submit only **one pdf copy** per each group with the naming format 'Assignment3_groupID.pdf'.

All homework assignments should have the student IDs and names of your team members. Remember that all homework assignments should be done in a group. This homework assignment should be submitted on EEE before 11:45 pm on the due date. Only one student in a group should submit the file. Everybody on the team is required to have the finally submitted version. Refer to the following table for the submission guidelines. After the 24-hour grace period, no more submission is allowed on EEE. That is, we will **not** accept assignments after that time. We will publish the solutions at that time for the next assignment. Please get all your work in on time!

Date / Time	Place	Remark
Thursday, April 28, 2016 11:45 PM	EEE dropbox	Due date
Friday, April 29, 2016 11:45 PM	EEE dropbox	24-hour grace period - 10 points will be deducted

Relational Algebra and SQL [100 pts]

Congratulations! Now, UCA has launched the database that you designed and start to serve passengers across the US. UCA prepares to hire some Data Scientists to investigate some interesting problems over the data. As an A+ taker in CS122A, you think these problems are easy. You decide to use what you've learned last week to tackle these problems.

UCA is using relations from HW2. Due to the limitation of our analysis platform, all the "datetime" attribute types are replaced with a "string" type. This change is only temporary for this work (HW3).

Schema (All "datetime" types are changed to "string" types only for this homework)

1. **Airplane** = {registration_number:string, model_number:string, purchased_year:number, manufactured_year:number, capacity:number}
2. **Airport** = {IATA_code:string, name:string, airport_city:string, airport_state:string}
3. **Credit_Card** = {cid:number, card_number:string, expr_date:string}
4. **Customer** = {cid:number, ssn:string, gender:string, email:string, address_street:string, address_city:string, address_state:string, address_zipcode:string}
5. **Customer_Reserves_Flight** = {cid:number, flight_number:string, projected_departure_datetime:string, purchased_datetime:string, purchased_price:number, quantity:number}
6. **Dish** = {lid:number, name:string, price:number}
7. **DishOrder** = {oid:number, cid:number, lid:number, order_datetime:string, total_amount:number}
8. **DishOrder_Contains_Dish** = {oid:number, lid:number, name:string, quantity:number}
9. **Flight** = {flight_number:string, projected_departure_datetime:string, projected_arrival_datetime:string, aiplane_registration_number:string, departure_airport_IATA_code:string, actual_departure_datetime:string, arrival_airport_IATA_code:string, actual_arrival_datetime:string}
10. **FlightAttendant** = {faid:number, phone_number:string, birthdate:date, ssn:string, job_title:string, address_street:string, address_city:string, address_state:string, address_zipcode:string, service_year:number}
11. **Lounge** = {lid:number, location:string, airport_IATA_code:string}
12. **MaintenanceEngineer** = {meid:number, phone_number:string, birthdate:date, ssn:string, job_title:string, address_street:string, address_city:string, address_state:string, address_zipcode:string, skill:string}
13. **OperationStaff** = {osid:number, phone_number:string, birthdate:date, ssn:string, job_title:string, address_street:string, address_city:string, address_state:string, address_zipcode:string, department:string}
14. **Pilot** = {pid:number, phone_number:string, birthdate:date, ssn:string, job_title:string, address_street:string, address_city:string, address_state:string, address_zipcode:string, since:number}

Part A. Relational Algebra [70 pts]

There is an online relational algebra calculator available at <http://138.232.66.66/ra/calc.htm?lang=en>. Follow the instructions at [HERE](#) to load sample records to the service. For each of the following queries, write a relational algebra expression. In addition, run the expression using the service, and paste the query results. Here's an example:

Q: Find IATA_Codes of all Airports.

Expression: π IATA_code (Airport)

Result:

Airport.IATA_code
JFK

LAX
SAT

1. [5pts] Find all female customers who live in Irvine.
2. [5pts] Find the names of dishes served by the lounge with an "lid" value 113.
3. [10pts] Find the credit card number and its expire date of a customer (cid:16).
4. [10pts] Find the gender, address_street, and address_city of a customer who ordered "wafu steak". Note that many lounges can serve "wafu steak".
5. [10pts] Find the name and quantity of each dish ordered by a customer who reserved only one ticket for the flight N124 on 08:21:00 Sep. 07, 2015(We use 'yyyy-MM-DD hh:mm:ss' for the string datetime format).
6. [10pts] Find the cids and emails of customers who haven't ordered any dishes at all.
7. [10pts] List the ids and ssns of customers who placed an order with every lounge (using Division operator).
8. [10pts] Find the ssns of employees from New Jersey.

Part B. SQL Queries [30 pts]

In this part, use SQL queries to do analysis on the dataset above.

1. Find cids of customers who reserved flights.
2. Find customers whose street address ends with "Warner Avenue".
3. Find customers who reserved flights and ordered dishes.
4. Find the locations of lounges which serve "tacos" or "hummus".
5. Find cids of customers who didn't order any dishes nor reserve any flights.