

Homework 2: E/R Models and More SQL
(due February 17th, 2016, 4:00pm, in class—hard-copy please)

Reminders:

- a. Out of 100 points. Contains 4 pages.
- b. Rough time-estimates: 4~6 hours.
- c. Please type your answers. Illegible handwriting may get no points, at the discretion of the grader. Only drawings may be hand-drawn, as long as they are neat and legible.
- d. There could be more than one correct answer. We shall accept them all.
- e. Whenever you are making an assumption, please state it clearly.
- f. Unless otherwise mentioned, you may use any SQL/RA operator seen in class/in textbook.
- g. Unless otherwise specified, assume set-semantics for RA and bag-semantics for SQL.
- h. Feel free to use the linear notation for RA and create intermediate views for SQL.
- i. **Important:**
 - a. For E/R diagrams, use only the style/notation/material given in the lecture slides.
 - b. A useful tool for creating E/R diagrams: <http://logicnet.dk/DiagramDesigner/>. You may have to manually draw-in some things though (like adding proper constraints etc.). There are other such good programs/sites too (like <http://creately.com/>, <http://visio.microsoft.com/>).
Note: Microsoft Visio is a commercial tool.
- j. Lead TA for this HW: Sorour Amiri.

Q1: Recommending Collaborators [20 points]

Consider ResearchBook, an online social network for researchers, which also helps one find possible collaborators. We are given a simple table with schema:

`ResInt (pid, topic)` - Person with `pid` has `topic` as a research interest

Clearly a person can have multiple research interests (like ‘Databases’ and ‘Artificial Intelligence’)—so there may be multiple tuples for each `pid` in the relation. The goal of this question is to write a SQL query that gives us pairs of researchers with exactly the same research interests (as we can recommend them as collaborators). This query is tricky—so let’s try to solve it by the following steps.

- Q1.1. (10 points) Write a SQL query to find all person-person-interest triplets (`pid1, pid2, topic`) such that person `pid1` has `topic` as a research interest, but person `pid2` *doesn't*. Call this the `PPT` view.

- Q1.2. (10 points) Given the earlier view PPT and the original table ResInt, write the full SQL query to find all pairs of people with the exact same set of research interests. Remove mirror pairs and self-pairs.

Q2. Where Art thou? [15 points]

Consider the situation in Exercise 2.8 in your textbook. We repeat it here for your convenience. Although you always wanted to be an artist, you ended up being an expert on databases because you love to cook data and you somehow confused “database” with “data baste”. Your old love is still there, however, so you set up a database company, ArtBase that builds a product for art galleries. The core of this product is a database with a schema that captures all the information that galleries need to maintain. Galleries keep information about artists, their names (which are unique), birthplaces, age, and style of art. For each piece of artwork, the artist, the year it was made, its unique title, its type of art (e.g., painting, lithograph, sculpture, photograph), and its price must be stored. Pieces of artwork are also classified into groups of various kinds, for example, portraits, still-lives, works by Picasso, or works of the 19th century; a given piece may belong to more than one group. Each group is identified by a name (like those just given) that describes the group. Finally, galleries keep information about customers. For each customer, galleries keep that person’s unique name, address, total amount of dollars spent in gallery (very important!), and the artists and groups of art that the customer tends to like.

The ER model your DB engineer designed was lost, and he has resigned. So you need to step in:

- Q2.1. (10 points) Draw an ER diagram for this database. Make sure to indicate primary keys, cardinality constraints, weak entities (if any), and participation constraints. List any assumptions you make in the process.
- Q2.2. (5 points) Translate the ER diagram in Q2.1 into relational database tables (i.e. give the SQL DDL statements). Make sure that the translation captures key constraints (primary keys and foreign keys if applicable) and participation constraints in the ER diagram. Identify constraints, if any, that you are not able to capture.

Q3. Hospital DB [20 points]

We want to design a database schema for a hospital. The patient is admitted to a hospital with a medical condition. The hospital maintains patients’ information. Name, age, sex, DOB, and address are recorded. Hospital identifies each patient by a unique id and creates a patient admission record. Each admission record has an admission number, admission date, and discharge date information. Hospital assigns a doctor to treat a patient. Hospital stores doctor’s name, id, specialty and year of experience. Each

doctor has multiple patients. A patient is admitted to a ward. Ward is identified by ward number, name, and type (e.g., medical/surgical). Each ward contains multiple beds. Bed has number and type (e.g., side room bed/ open ward bed) as well. If patient needs a surgery than hospital schedule an operation. Operation number, date, time, patient id and doctors' id are required to schedule an operation. We assume multiple doctors participated in an operation. Draw an ER diagram based on above scenario. List all of your assumptions you make for your drawing.

Q4. Real-estate Database [25 points]

We are asked to design a database management system for all information related to a real-estate company which has several branches throughout the United States. The first step is to organize the information given about company. We have collected the following data:

- Each branch has a unique branch number. It allocated staff, which includes one Manager. Also, each branch has a list of available houses to rent/sell.
- The manager is responsible for the day-to-day running of a given branch.
- Each staff has a unique staff number, name, position, salary, and branch number.
- Each house has a unique house number, address, rent cost, sell value, status, branch number.
- The statue of a house indicates whether it is available for rent/sell. The branch number indicate which branch of the real-estate company can rent/sell the house.
- A Customer has SSN, name, contract number, house number.
- Each Contract has a unique contract number, type, date, branch number, and customer number.
- If the type of a contract is rent, it will have a deposit, and rent cost, and lengths of contract. The length of a rent contract can be six months, one year or 2 years but costumers can extend their contract by signing a new contract.
- Each contract extension has a unique extension number, the original contract number, and discount value.
- If the contract type is "sell", it will have sell value.
- Customers can issue a contract termination request. Each termination request has a unique request number, customer number, and termination cost.

Please answer the following questions:

- Q4.1. (15 points) Draw an ER diagram for this database. Make sure to indicate primary keys, cardinality constraints, weak entities (if any), and participation constraints. List any assumptions you make in the process.
Hint: You may need an ISA hierarchy somewhere.
- Q4.2. (10 points) Translate the ER diagram in Q4.1 into relational database tables (i.e. give the SQL DDL statements). Make sure that the translation captures key constraints (primary keys and foreign keys if applicable) and

participation constraints in the ER diagram. Identify constraints, if any, that you are not able to capture.

Q5. Subject Code Formatting [20 points]

Suppose in a university database there is a table called "Subjects" with a single unique "code" column. Assume you are given the following SQL script:

```
CREATE TABLE Subjects
  (code VARCHAR(15) NOT NULL PRIMARY KEY);
INSERT INTO Subjects
VALUES ('CS'), ('ECE'), ('EDCO'),
      ('ART'), ('SOC'), ('MKTG'),
      ('STAT'), ('BC'), ('PHYS'),
      ('FST'), ('FIN'), ('SPAN'),
      ('ARCH');
```

Now the "SELECT code FROM Subjects ORDER BY code" query will return the subject codes in alphabetic order. However, we would like to organize the list in two columns. This is tricky. Let's first answer a simpler query instead.

For the questions below, make sure both your SQL queries can run "as-is" in SQLite. Use ".header on" command to see column header(s) in SQLite.

- Q5.1. (10 points) Write a SQL query to create a VIEW called "ALTSKIP" which displays a version of the Subjects table by skipping each next entry. Arrange your list in alphabetic order. For example, on our dataset, your output should look like following:

| |
|--------|
| FirstC |
| ARCH |
| BC |
| ECE |
| FIN |
| MKTG |
| SOC |
| STAT |

Hints:

1. Of course your query should be general, and be able to run on any instance of the Subjects table, not just the one shown above.

2. You maybe need a self join and a Group by and Having.
3. The modulus operator in SQLite is % (other systems have other syntax): $x \% y$ will give you the remainder after dividing x with y . You may find the following SQL fragment useful: `SELECT (COUNT(code)% 2) FROM Subjects` will give you 1 if the total number of subjects is odd; 0 otherwise.

Q5.2. (10 points) Using the view you made in Q5.1 ("ALTSKIP"), write a SQL query that will display subject codes across in two columns like the following:

| FirstC | SecondC |
|--------|---------|
| ARCH | ART |
| BC | CS |
| ECE | EDCO |
| FIN | FST |
| MKTG | PHYS |
| SOC | SPAN |
| STAT | |

Hint: You may need a left outer join. Again, your query should be general and be able to run on any instance of the Subjects table, not just the one shown above. The empty cell can be NULL as well.