

CS 146: Homework 2

Due Feb.1st, 11:59pm. **No extensions allowed on this homework.**

You can submit either (a) hardcopy in department mailbox by 5pm Friday Jan.30th or (b) on blackboard by 11:59pm.

The database you will use for the first three problems of this assignment contains information about people, vacation locations, and when those people visited them. The relations are:

- Person(ID, Name)
- Location(PlaceName, Country, MainAttraction)
- Visited(ID, PlaceName, Year).

You should assume that place names are unique in the world.

Ques.1: Consider the schema given above. (a) Give a primary key for each relation. Are there any relations for which there is an alternate candidate key which you have not chosen as the primary key? Why or why not? (b) State all referential integrity constraints (inclusion dependencies) that should hold on these relations.

Ques.2: Write relational algebra queries for each of the queries below. In addition, write tuple relational calculus queries for at least the first three queries. If a query is long, feel free to break it up into a series of queries with intermediate answers stored using the assignment operator in temporary relations (e.g. $r \leftarrow x$). Note: Your queries must not be dependent on the instance of the database (i.e., they should work without modification even if another instance of the database is given) and ignore issues of duplicates.

- Return all tuples in Location where the main attraction is a beach.
- Return all the places visited by someone before 2000.
- Return the names of everybody who visited somewhere in 2003.
- Return pairs of names of people that visited the same country in the same year. The result should have schema (Name1, Name2).
- Return the names of the unfortunate people who didnt visit anywhere at all.
- Return the names of the lucky people who visited all of the places listed. You may assume that every PlaceName in Location occurs in Visited.

Ques.3: For the aforementioned schema, what does the following query compute: $\pi_{PlaceName}(Location) - \pi_{PlaceName}(\sigma_{Year > 2003}(Visited))$.

For the next two questions (Questions 4,5) you will use a database that stores information on employees and companies they work for. The schema has the following tables, whose SQL definition is also provided:

- EMP(SSN, Name, Birthdate, Street, City, DNO, SuperSSN, Salary) – stores information about employees, with SSN as key. The DNO denotes the department to which they are assigned and the SuperSSN is the SSN of their supervisor. Every employee has a supervisor (assume the CEO has a supervisor with a fictitious SSN).

- DEPT(DNUM, Dname, MGRSSN) – stores information about the departments such as the unique department number, a unique department name (Dname), and the SSN of the department manager (every department must have a manager).
- PROJECTS(PNum,Pname, PLocation, DNO) – information about projects at the company, with unique Project number (PNO). The project has a name and a location and is controlled/managed by a department.
- WORKS(ESSN, PNO, Hours) – information about the projects that each employee works on and the number of hours they work on each project.

Ques.4: Write relational algebra queries for the following:

- Find the names and addresses (street and city) of employees who work for the the Research department.
- For every project located in Stafford, list the project number, the controlling department number and the manager's name.
- For each employee, find their name and their supervisor's name.

Ques.5: For each of the relational algebra queries below, describe what the query computes.

(a): $\pi_{SSN}(EMP) - \pi_{SuperSSN}(EMP)$

(b)

$$\pi_{W2.ESSN}(\sigma_{(WORKS.pno=W2.pno)}(\pi_{WORKS.pno}(\sigma_{(name='Smith') \wedge (ESSN=SSN)}(WORKS \times EMP) \times (\rho_{W2}(WORKS))))))$$

```

CREATE TABLE EMP (
    SSN int PRIMARY KEY,
    Name Varchar(50),
    Birthdate Date,
    Street Varchar(50),
    City Varchar(50),
    DNO int,
    SuperSSN int,
    Salary Real,
    FOREIGN KEY (DNO) REFERENCES DEPT(DNUM),
    FOREIGN KEY (SuperSSN) REFERENCES EMP(SSN));

CREATE TABLE DEPT (
    DNUM int PRIMARY KEY,
    DNAME Varchar(50),
    MGRSSN int,
    UNIQUE DNAME,
    FOREIGN KEY (MGRSSN) REFERENCES EMP(SSN));

CREATE TABLE PROJECTS(
    PNum int PRIMARY KEY,
    Pname Varchar(50),
    Plocation Varchar(50),
    DNO int,
    FOREIGN KEY (DNO) REFERENCES DEPT(DNUM));

CREATE TABLE WORKS(
    ESSN int,
    PNO int,
    HOURS int,
    PRIMARY KEY (ESSN,PNO),
    FOREIGN KEY (ESSN) REFERENCES EMP(SSN));
    FOREIGN KEY (PNO) REFERENCES PROJECTS(PNUM));

```