CS 338 – Winter 2014
Assignment #3

(Solutions will be posted on March 7th)

Part I

You have been asked by the Waterloo Recreational Sports Association (WRSA) to help them with their database design. The WRSA has come up with a database specification and included it in the next section. You will be required to create an ER model based on the specification. Once you have created the ER model you will map it to a relational model using the steps from lecture.

Database specification:

The WRSA oversees several sports. Each sport has a unique name, the number of referees required to referee matches of that sport, and a list of locations where the sport can be played. For simplicity, each location is a single string. The WRSA is comprised of many teams. Each team plays exactly one sport, and every sport has some team associated with it. Each team has a unique name. A team may choose to have one or more coaches. However, since the WRSA is recreational, teams are not required to have a coach. Coaches have a name, address, phone number, and certification date. Players who are registered in the WRSA are members of teams. Each player can be a member of multiple teams, but they must be a member of at least one team. Players have a name, age, phone number, and address. Coaches and players are covered by the WRSA's insurance and each have an insurance number. Each player must have at least one emergency contact. The emergency contacts each have a name, address, relation to the player, and phone number. A player cannot be a coach and cannot have two emergency contacts with the same name. The WRSA selects several players to act as mascots. Each mascot must have a nickname. Every team plays in matches with other teams. A match is given a unique match number, a location, and a date and time. Matches are always between two different teams and the outcomes are not recorded. Each match is refereed by other players who are not involved in that specific match.

Question 1:

Create a ER model based on the database description. Make a registrant entity that is a superclass of players and coaches. Give registrants unique IDs.

For each entity you create, ask yourself:

- What are its attributes?
- What attributes can make up its key?
- Is the entity weak?
- Is the entity a subclass of another entity?
- What relations does it have with other entities?

- Are any attributes multivalued?

For each relation you create, ask yourself:

- What is its cardinality ratio?
- What are the participation constraints?
- Is it an identifying relationship?

Come up with at least one derived attribute and include it in your ER model.

Question2:

Create a relational model based on your ER model. Make sure to follow the six steps outlined in your lecture notes.

Part II:

Answer the following questions based on this schema:

Emp(*eid*: integer, *ename*: string, *age*: integer, *salary*: real)
Works(*eid*: integer, *did*: integer, *pct_time*: integer)
Dept(*did*: integer, *budget*: real, *managerid*: integer)

And suppose you have two views defined as following:

CREATE VIEW SeniorEmp (sname, sage, salary)
      AS SELECT E.ename, E.age, E.salary
          FROM Emp E
          WHERE E.age > 50

CREATE VIEW AvgSalaryByAge (age, avgSalary)
      AS SELECT E.eid, AVG (E.salary)
          FROM Emp E
          GROUP BY E.age

Question 1:

Write an SQL query, <u>without using any views</u>, to produce the same results as the query below:

SELECT S.sname
FROM SeniorEmp S
WHERE S.salary > 100,000

Question 2:

For each of the above two views: can you insert a new record into the view? (If the answer is Yes, give the corresponding SQL statement. And if the answer is No, give the reason for the failure.)

Question 3:

What would be the result of executing the update statements below?
1) UPDATE SeniorEmp
   SET salary = 110,000
   WHERE salary = 100,000

2) UPDATE AvgSalaryByAge
   SET avgSalary = 110,000
   WHERE avgSalary = 100,000