Introduction to Database Design, E2015
IT University of Copenhagen
Carsten Schürmann

Homework 1: Database design

Theoretical Part

Problem 1:

Consider a database to store information about a Research Organization.The database has the following properties:

- Every department has a title, and a unique department ID (departmentID).
- A department may have zero or more employees.
- Each employee belongs to exactly one department. We store the name of the employee and a unique employee ID for each employee (employeeID).
- Employees can be researchers or managers. For managers we store their annual bonus amount and for researchers their doctorate degree subject
- People work in projects. Each Project has a unique project ID (projectID).
- Every project has exactly one manager and zero or more researchers.
- A manager can manage one or more projects but a researcher must work on exactly one project.

Given this description of the database and its constraints, we have created a mostly correct Entity-Relationship Diagram, shown in Figure 1.

Question 1: (10 points) Find and correct any mistakes in the given ER diagram. Specifically, number and list them, like, e.g.

1. delete: arrow, from x to y
2. change to bold line: thin line, from z to w
3. change to bold box: entity e

Question 2: (15 points) There may also be some missing element(s). If none, say none - otherwise, add them to the picture, and list them, numbered. E.g.

1. add: attribute a, to entity e
2. add: bold line, arrow, from c to d.
3. add: weak entity, f, with attributes ....

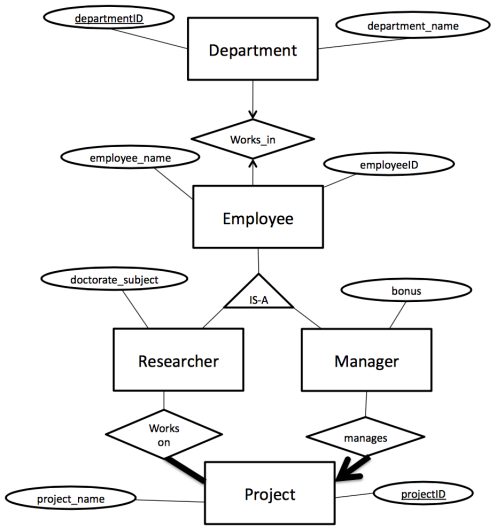

Figure 1: ER diagram with bugs

Question 3: (10 points) List and number all the bold lines and all the arrows that are in the final, corrected version of the diagram. E.g.

1. bold, line, from Department to Employee
2. thin, arrow, from x to y

Hints List your assumptions, if any. We will accept all reasonable assumptions.

Problem 2:

Consider a database for a Copenhagen art gallery. It records information about artists, paintings and exhibits. The constraints are exactly as shown in Figure 2. Paintings, Artists and Exhibitions have unique identifiers as shown in the Figure, with binary relationships among them as illustrated. To clarify:

- The line from Exhibition to showcases, is thick.
- The arrow from Painting to paints, is also thick.
- No other lines, boxes, or diamonds, are thick.

Question 4: (20 points) Give the DDL statements, that correspond to the above ER diagram.

1. Use proper data types (well accept all reasonable choices).

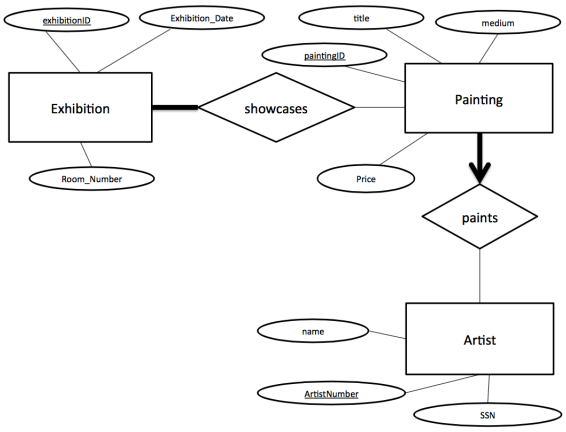

Figure 2: ER diagram for art exhibits

2. Avoid syntax errors (well forgive missing semicolons).
3. Specify your decisions with respect to CASCADE deletions. (E.g., I decided to reject deletions in Exhibition, when there are still participating artists)
4. Without using CHECK statements, enforce as many as possible of the implied integrity constraints as you can.

Question 5: (5 points) Which of the implied integrity constraints of Figure 2 need CHECK statements to be enforced? List them all, or say none. For example, a (possibly, correct) answer could be: *thin line, from Painting to showcases.*

Problem 3:

Question 5: (6 points) Consider the following legal instance of a relational schema S with attributes ABC:

A	B	C
α	9	T
α	16	T
β	20	F

1. Which of the following dependencies are violated by this instance of S? Answer with yes or no.
   (a) $(A \to B$
   (b) $B \to A$

(c) $C \rightarrow A$
(d) $AC \rightarrow B$
(e) $B \rightarrow AC$

2. By only observing the instance of S above, can you identify the functional dependencies that hold on schema S?

Question 6: (19 points) For the next set of questions consider the relational schema $A(P, Q, R, S, T, U, V)$ and the set of functional dependencies FD:

$$\begin{aligned} P &\rightarrow Q && (1)\\ Q &\rightarrow R && (2)\\ PS &\rightarrow TRV && (3)\\ QT &\rightarrow UR && (4)\\ S &\rightarrow V && (5) \end{aligned}$$

1. Which of the following is a minum cover of the FD?
   (a) The given FD is a minimum cover.
   (b) $\{P \rightarrow Q, Q \rightarrow R, PS \rightarrow T, QT \rightarrow UR, S \rightarrow V\}$
   (c) $\{P \rightarrow Q, Q \rightarrow R, P \rightarrow T, Q \rightarrow U, S \rightarrow V\}$
   (d) $\{P \rightarrow Q, Q \rightarrow R, PS \rightarrow T, QT \rightarrow U, S \rightarrow V\}$
   (e) None of the above. Give the right answer.
2. Which of the following functional dependencies can be deduced, from the above set of functional dependencies (Eq. (1)-(5))? Answer yes or no.
   (a) $P \rightarrow R$
   (b) $PS \rightarrow U$
   (c) $QS \rightarrow U$
   (d) $QST \rightarrow P$
3. True or False: The attribute closure $\{Q\}^+$ is $\{Q, R, T\}$.
4. True or False: The attribute closure $\{PS\}^+$ is $\{P, Q, R, S, T, U, V\}$.

Practical Part

In this project, which consists of three parts, you are going to work with a movie database. The first step is to make a suitable data model, documented by an E-R diagram and implemented as an SQL schema definition. Later on you will be working with actual data about movies. An anonymous student suggested the following relation schemas for some of the information:

```
Person(id,name,gender,birthdate,deathdate,height)
Movie(id,title,year,color,country,language,imdbRank)
Contract(personId,movieId,isActor,isDirector)
```

To extend this to a more comprehensive data model, use The Internet Movie Database (IMDB), www.imdb.com, for inspiration. The database should store information about actors/actresses, directors, writers, movies, genres, awards, ratings, etc. Your design should incorporate the above to the extent possible. In your initial design, there is no need to worry about database performance or other things. Note: IMDB stores lots of information. You cannot include all types of data in your model, hence you have to decide which information to include and which to exclude. Try to choose the most interesting data, both in terms of the application area and in terms of the model. E.g. it is not necessary to have 10 attributes describing aspect ratios and other technical details of a movie. Feel free to include additional information not found in IMDB, if you like. The final relational model should not include more than about 10-15 tables and 5- 10 tables can be enough, depending on your choice of tables and what kind of data you choose to store in your database. You should aim to create a design that is in normal form (BCNF).

Problem 4:

Question 7: (10 points) Devise a model of your database in the E-R notation of RG. Remember to include all attributes, cardinality constraints, and keys. You probably have to make assumptions about the data. Include a description of the assumptions and choices you have made.

Question 8: (5 points) Give a corresponding schema for your movie database (see RG, page 59) This should be what you get by following the method for translating an E-R model to a relational one.

Question 9: (5 points) Translate the relational schema into the SQL/DDL syntax, You should make sure that your schema is accepted by the MySQL DBMS. (Install MySQL on your own machine using the links on the course home page, or create an account at mysql.itu.dk.) The schema must include

- Names of relations,
- column names and types,
- primary keys and possibly other candidate keys, and
- foreign keys constraints
- all other constraints that you deem necessary to guarantee the continuous integrity of your database.

Question 10: (10 points) Give an analysis of keys and functional dependencies in all your relations. This should be detailed enough that it would convince another student who knows about normalization that your design is in normal form. For example, you could address the top-10 reasons why someone might doubt that your design is in normal form. If you end up with a design that is not in normal form, go back and modify your E-R model to correct this.

The first hand-in should be handed in by each group no later than

Wednesday 16.09.2015, 12:00.

It suffices that one group member sends the solution as a single PDF file with a file name that includes the group number. Please upload your reply to learn-it. Late hand-ins will not be considered. This project will be graded and contribute with 15% to your final grade.