

## Database Design Project

NOTE, this is a team project. Up to three students constitute a team. STUDENTS MUST EMAIL THE FINAL REPORT TO Dr. YU ([byu@udc.edu](mailto:byu@udc.edu)) BY MAY 7, 2013 5:00PM.

The purpose of this project is to give you some experience in the design of a database. We will explore both theoretical and practical aspects. The emphasis is on the theory of logical database. Once you master it well, it is a simple matter of translating the theoretical concepts you developed into the terms of a concrete DBMS to implement the database.

### Part I: Conceptual/Logical Database Design

You must choose some facet of real life that you feel needs to be modeled by a database. Then, you must design a relational database schema that adequately represents this information, and demonstrate its efficacy by providing a set of useful queries that your database can support. In the process, you must use standard relational modeling tools and techniques we examined in class. In concrete terms, this part of the project consists of the following steps:

- (1) **Begin by choosing an application**, e.g. database for a local volleyball league or for a Society of Roasted-Shrimp Fans. It must be non-trivial, but not huge. As a general guideline, you should probably have about seven to ten entity sets and similar number of relationship sets.
- (2) **Formulate in English at least 10 realistic queries**, which will often be posed on the database. This means queries you believe would be useful to somebody using the data. The queries must be realistic and you should have more than a few complex queries. Note, the queries that the database system must be able to answer ultimately determine what information needs to be maintained in the database. For example, you need not include the information about best friends of students if no user of your database will be asking for that. On the other hand, some additional information, not reflected in your 10 queries, may be included in the database if you feel that somebody may be interested in it.
- (3) **Specify the assumptions about the database in English (informally)**. Here you talk about attributes, keys, the nature of relationships between entities, etc. Do not discuss something that is obvious (e.g., that a student can take several courses). In addition, don't make too many simplifying assumptions.
- (4) **Produce an ER diagram** for your database, which will reflect your choice of the entity sets, their relevant attributes, and the relationships among them. Again, make your relationships realistic and meaningful.
- (5) **Convert the ER diagram into a relational database schema.**
- (6) **For each relation schema obtained in Step (5), specify the functional dependencies** satisfied by the schema. Based on the knowledge of functional dependencies, you can analyze how good your initial DB design is, i.e. the potential for repetition of information and other anomalies we examined in class. For each relation schema, specify the normal forms it satisfies.
- (7) If necessary, **perform decompositions of the relations**, such that the entire database is in 3NF or BCNF. Note that if your initial design is non-trivial, then at least one table should be in 1 or 2NF, so that you can decompose it.
- (8) **Formulate the queries in Step (2) using relational algebra**. If, at this point, you are unable to state the queries in relational algebra, this is probably because your database is not powerful enough. Then, you must go back to step 3 and see how the database can be improved.

- (9) Of course, no database is capable of supporting all possible queries. Very often (in fact, it is almost always the case), databases are unable to support some reasonable ad hoc queries. I expect that your database will be one of them. Thus, to evaluate the power of your database properly, you must identify reasonable, but unanswerable queries as well, and explain why your database cannot support them. For the purposes of this project, it will be sufficient to state **two** queries of this kind in English.

## **Part II: Report on Logical Database Design**

In practice, DBAs must prepare documents for her/his peers, superiors, successors, and the database users.

You should hand in a full legible report (typeset) with a cover page that gives your name, the title of the project, and its brief description. The final report should include sufficient detail to describe all steps of your design:

- (1) Describe what you have chosen as your database.
- (2) List at least 10 reasonable queries in English.
- (3) Add one or more atomic transactions to your query set.
- (4) State assumptions about the database as briefly as possible.
- (5) Give the ER diagram of the database.
- (6) Give the relational database schema.
- (7) For each relation schema, give the associated FDs and specify the normal forms the schema satisfies.
- (8) Perform the decomposition into 3NF or BCNF and give the final database schema along with the FDs.
- (9) Give the relational algebra expressions for the queries formulated in step 2.
- (10) Give at least two reasonable, but unanswerable queries, and explain why the database cannot support them.
- (11) In conclusion, describe what you have learned from the project, and include your observations about the power of relational systems and languages.

## **Part III: Implementation**

In general, after the logical database design is performed, the DBA (or application developer) must implement the database system using a real DBMS. In other words, the administrator must use the DDL (e.g., the DDL features of SQL) of the DBMS to define the relations and constraints. In addition, using a query language supported by the DBMS (e.g., SQL), the administrator typically implements some common queries on the database.

Your task now is to implement your database system using Oracle or MySQL. For that,

- (1) Create tables obtained in Step (7) of Part II.
- (2) Insert 10 to 15 records into each table and print the contents of each (using SELECT \* ).
- (3) Compose SQL queries formulated in Step (8) of Part II; Add **one or more atomic transactions** to your query set; Execute the queries and print the results.
- (4) Append the printouts (i.e., *create table* statements, *insert into* statements, *select* statements (queries), and their results) to your report on logical database design of Part II of this project.