

CS5035: Introduction to Database Systems **Assignment – Database Design and SQL**

Date to be handed in: 3pm Friday 3 December 2010

This exercise counts as 25% of the overall course assessment. In this assignment you design a database for the following user requirements and run queries on your designed database. This exercise is part of the formal assessment of the course, and the work done must be your own. You are reminded to read the section on Cheating and Plagiarism in your student handbook. From the CS5035 assessments page, download the CS5035Assignment.doc which you should use to submit your answers.

User Requirement Specification

A local primary school approaches you to develop a database for managing data about their sports day to be held in spring 2011. Please note that as a database designer you should learn to work with incomplete, ambiguous and perhaps even inaccurate requirements. Often designers use their understanding of the domain to supplement information available from the client.

Data Requirements:

Every year children participate in the following events:

1. 50m running race
2. 200m relay
3. Football
4. Rugby

All the children are required to participate in at least two games. Running race and relay are conducted separately for each class. For relay, children in a class are invited to form several six member teams. For football and rugby children in the school are divided into four age groups, P1-P2, P3-P4, P5-P6 and P7. Children in each age group are asked to sign up for one of the two football teams in that age group. Similarly children in each age group are also asked to sign up for one of the two rugby teams in that age group. Boys and girls both participate in each of the teams. If any age group has lesser children than required to form two standard teams, they are allowed to form teams smaller than the standard but equal in size. In each age group the two teams play against each other. Each team is given a name. Each of the events requires some equipment such as goal posts and nets. Each of the events needs to be supervised by a member of staff. Some children in the school may not be medically fit for participating in some of the events. The school is interested in only recording the winner in each event rather than the detailed scores. However, a comment about each game is required where the detailed scores and any other details about a game can be recorded.

Transaction Requirements:

1. List all the events and their scheduled times.
2. List all the winners and the corresponding event details.
3. List the staff supervising a given event (assume the given event).
4. List all the equipment used in an event.
5. List all the children participating in an event.
6. Count how many relay teams are formed from each class
7. Count all the events participated by each student and sort the list.
8. Add a team to a given relay event.
9. Change the time of an event
10. Move a given staff from supervising one event to supervising another

Design a database for the above requirements first by creating an EER model and then deriving a relational model from the EER Model. Make sure that all the tables in your design are in 3NF. Implement the database on the MYSQL server by creating a new database using the procedure described in practical 3. (Since you have already created a MYSQL account, you only need to create a new database). Insert at least 3 rows of data in all the tables. If you are not happy with your design, go back and fix the problems. It is a good idea you learn this iterative approach to database design. Write standard SQL queries to fulfill all the transaction requirements stated above assuming any data you may need. The transactional requirements are not precisely specified and you may have to

make assumptions which you should state clearly in your submission. If any specific requirement cannot be fulfilled, check if this is because of your specific design.

For Submission

- a) Create an (E)ER diagram showing your conceptual design for the above requirements. Please assume that your ER diagram will be used by someone other than you for carrying out the relational design. To help the new designer, your ER diagram should contain all the required information. This means, your ER diagram
 - i. contains meaningful labels for all the entities, attributes and relationships.
 - ii. shows all the required relationships and their attributes. (Note: Not only entities but also relationships have attributes. In UML notation attributes of a relationship are shown in a rectangle linked to the relationship using a dotted line.)
 - iii. marks the multiplicity constraints for all the relationships.
 - iv. contains textual annotations to explain your design decisions which you feel might be useful at the relational design stage.

Paste your ER diagram into the submission template. Please state clearly any assumptions you make. (6)

- b) Pick out any one table (with more than three columns and having a multi-column primary key) from your database and explain how it is in the third normal form (3NF). (Please note that to show that a table is in 3NF you have to first show that it is in 2NF which in turn requires you to show that it is in 1NF.) (3)
- c) Your newly created database can be exported into an SQL script. In the PHPMYADMIN interface select your database and go to the Export tab. Accept all the defaults to export the database. The system displays the SQL script for your database. Paste all this information into the submission template. (6)
- d) For each of the transaction requirements cut and paste your SQL statement and the result table in the submission template. (10)

Create a zip file with your submission template and mail it to yaji.sripada@abdn.ac.uk. In the subject field type only 'A for CS5035' and nothing else. Please follow all the submission instructions to help us mark your coursework efficiently.