1 Revision Questions

Lecture 5 was about entities and business rules. In this tutorial you are to look at three mini case studies, and determine

- What entities are involved in each case
- What business rules are clear from each case
- What might be needed to be known to complete an understanding of the business rules for each case

The cases are:

1. A network manager in a medium sized corporation wants to have a database of all the machines in the organization, in part so that she can identify who are the users who might log on to each machine. She also wants to know what type of machine each is [including some of its characteristics, for example, brand, amount of RAM, amount of secondary storage, operating system, applications loaded, etc], what monitor each system has, any other characteristics. The organization has 125 LANs, and each LAN can have up to 252 machines, although this would severely load that LAN, and is not typical – most LANs have a maximum of about 100 machines, and many LANs have as few as 25 machines. The machines are UNIX systems, some IBM style personal computers and some Apple iMacs. Some of the machines are laptops, both IBM and Apple. Each LAN has a server. The laptop machines can, of course, be plugged in to any [free] configured port on the LAN.

2. A motor vehicle fleet manager is keeping track of about 280 vehicles for a government department. Some of the vehicles are “company cars” allocated to individuals and driven by them, some are fleet cars, for use by the minister of the government department, the junior minister, and the department secretary. These vehicles are chauffeur-driven. There are also about 60 vehicles that are part of a pool of vehicles, used on a day to day [or even hour to hour] basis by staff of the department. There are also a number of miscellaneous vehicles – 6 four-wheel drive vehicles for accessing remote outposts of the department, two graders, 3 bulldozers, 18 trucks, 43 utilities and a low-loader. The fleet manager needs records of vehicle age, condition, mileage [or in the case of the heavy vehicles, hours of use], driver records, fuel usage, etc. [Remember, you may discuss other data that might be of use to capture.]

3. A video library lends videos to registered borrowers. The borrowers must always provide some proof of identity, preferably a borrower card from the
video library itself. Borrowers may have up to 2 overnight videos, and can also have up to 6 weekly videos. The usual hire charge for an overnight video is $6 and the hire charge for a weekly video is $1. Items returned late attract a penalty of $5 plus any additional rental fee. The manager of the video library tries to manage the stock, and to keep track of borrowers, and borrowed videos.

2 Using MS Access to Create a Simple Database

In this exercise we will build on and further develop the simple database that you started to create last week. As in last week’s tutorial, this exercise is aimed at encouraging you to explore and learn about MS Access by playing and experimenting. As we mentioned last week, MS Access has been designed with ‘learning by exploration’ in mind.

Joining tables with relationships

This week you will add to the database you started to create last week. By now you should have created and populated your two database tables with data about members and their ticket purchases.

1. Open your MS Access database so the Database window appears. Select ‘Tables’ and the two tables in your database should be listed
2. Select the ‘Tools’ menu, and then select ‘Relationships…’ - a screen showing your tables should appear
3. Create a relationship between the two tables by firstly selecting and highlighting the PK (Member No) field in the member table. Hold down the left mouse button on the highlighted field and drag the mouse to the corresponding field (Member No) in the ‘Tickets’ table (Member No is a foreign key in the ‘Tickets’ table).
4. The ‘Edit Relationships’ dialog box appears. Check the ‘Enforce Referential Integrity’ box and press ‘Create’.

A line representing the relationship now appears linking your two tables. The Member table is at the ‘1’ end of the relationship while the Tickets table is at the many (∞) end of the relationship. Your tables are now joined on the common data element Member No.

Sorting data in tables

5. Close all windows except the Database window. Double click on the Ticket table to display the datasheet view. Position the cursor in the first field of the first record and select the ‘Sort Ascending’ control from the menu bar. This sorts the record based on the field where the cursor is situated. Move the cursor to the next field of the first record and repeat the exercise to see how it changes the order of the data in the datasheet. Try again using the ‘Sort Descending’ control on the menu bar. Close the ‘Ticket’ datasheet.