

Monday, 7 December 2009

[Reflections on week 5](#)


There was quite a lot left to cover in the last lecture session on Friday (I really don't like having the 2 hours of lectures together in one slot), which was peer observed. The timing would have worked better if I had stuck to using the extra Monday slot beyond the first two weeks rather than be persuaded not to by the groups of students who had no other classes that day! Lectures on a Friday are not ideal anyway, particularly in a 6-week 100% coursework module where we are supposed to have all assessment completed within the semester - not much point in having a lecture on the very last day when courseworks are due to be submitted.

I thought the fact that most groups would be doing join queries in the labs before I covered this in the lecture would be a problem, but in fact the students generally worked through the exercises quite happily with reference to the notes. Maybe lectures don't make any difference!

Class test marks were not too bad, but probably not as good overall as in previous years. I haven't done a detailed comparison yet to back up this impression, but the module generally seems to have been a bit more of a struggle this year.

I made a few changes on the fly to notes and tutorials. Some students starting on the coursework were struggling with the implementation of a join table, and the notes describe how this works but do not give a full SQL example. The many-to-many relationship in lab 2 is implemented through the Hires table, which has a bit more in it than a basic join table. I added the SQL code for the example table in the notes. In the normalisation table, I changed the 3NF example to use the same example as 2NF except with a surrogate key in the starting table rather than a compound key. This seemed to me to be a better illustration than the existing example. I did it this way ad-hoc in the tutorials and changed the tutorial sheet for next time.

On the subject of normalisation, I took the explicit question on normalisation out of the coursework for this year. The nature of the question in previous years didn't sit well with the theme and practical nature of the task and was generally badly done. The marks for this are given to Peerwise now, and the table implementation task has a general requirement to "check that tables are normalised". It will be interesting to see if anyone writes questions about normalisation in Peerwise. Should normalisation be done in this module at all? It's not explicitly identified in the LOs. I think it should. However, this module will be running for INTO next semester, and the content may need to be reconsidered for that.

Posted by Jim Paterson at [01:37](#) [0 comments](#) 

Labels: [lectures](#), [normalisation](#)

Monday, 30 November 2009

[Reflections on week 4](#)

The main event this week was the class test, a written exercise involving designing a database schema based on a UML data model. This schema will then be used as the basis for creating a

database in the hand-in assignment, and makes sure that the students have a correct schema to work from. Haven't marked this yet, so no conclusions on the results until next week. Attendance at the classes where test was taken was marked higher than the norm!


Missed the Friday lecture slot due to DBDC meeting. There is still time to cover the remaining material in next Friday's lecture slot, but it would have been better to have had the extra Monday slot to push on a bit this week. Unfortunately, one of the groups persuaded me to agree to not use this after the first two weeks as they had no other classes that week. In future, it is worth remembering that in a six-week module it is important to keep the flexibility available as missing out one lecture slot can really throw the timing off. The result this year is that the labs in week 5 will be ahead of the lecture in terms of content.

Lab 4 was done within the same session as the class test by students who were finished lab 3. Observation suggested that a fairly high number of students attempted this. Lab 4 is an exercise which shows integration of a database and a simple Java application (well, a BlueJ project which allows interaction with objects). Got some quite good reaction to this on the 'reflection' task on Blackboard. However, fewer and fewer people are doing these reflections.

Peerwise - about 60 students have participated and there are nearly 200 questions in the system. One week to go!

I changed the approach to the tutorial ("errors in queries") as the week went on from discussion/board to discussion/live demo. This is the first time that all tutorial rooms have had projectors, and it proved very simple and effective to download the database and try out the queries interactively to confirm the conclusions of the discussion, and to show exactly what errors were produced. It may be worth thinking about the other tutorials to see whether they could be improved with live demos.

I looked again at chapter 3 of the notes, and altered the structure a bit: the section on dates was moved to its own heading as it was maybe a bit hard to find for reference. The original version showed a subquery then repeated this and filled in the details later on - this was a hangover from the fact that I was not originally going to put details of how to do subqueries in the notes and was just going to show a brief example of how they might be useful and leave it at that. This has been made more coherent with all discussion of subqueries within a single section.

Posted by Jim Paterson at [02:03](#) [0 comments](#) 

Labels: [class test](#), [notes](#), [peerwise](#), [timing](#), [tutorials](#)

Friday, 20 November 2009


[Reflections on week 3](#)

Having encouraged students to register for Peerwise in week 2, I have now announced participation levels required to get credit, with a deadline at the end of week 5. There have been a few students with problems with registering, but most students who have tried have got into the system successfully. One problem was my fault as I assigned the same ID to 2 students (IDs were assigned in the Blackboard Grade Centre, so they are available individually to students in My Grades - this avoided the need to distribute IDs individually by email or any other way). I am now getting a trickle of emails from students who have

forgotten usernames, etc, and can't get back on - not many, but these still have to be dealt with, for example by assigneing new IDs. There are still a significant number of assigned IDs which have not been used, so some students have not engaged. Attendance in classes is generally poor, so it is not obvious exactly how many students I should expect to be active on the module. About 20 questions have been contributed so far, and these are generally relevant and some of them are quite good. I expect to see a surge in activity as the deadline approaches.

Tutorial 3 was an exercise in devising a schema based on a data model, as preparation for next week's class test. The sheet has a main question on that topic, and a couple of other questions. In previous years I think I have had time to cover the whole sheet in one session, but this has not been possible this year. There has been some variation between groups, but the number of students who come into the tutorial with some idea of how to implement relationships was worryingly low, despite this having been covered in the lecture the week before. The solution to the tutorial exercise was given in the tutorials and will be made available on the website. As this exercise is similar to the class test, students will hopefully study this...

Lab 3, SQL queries, was reasonably straightfoward as usual, and students generally work through it quite happily. I'm a bit uncomfortable with the fact that there wasn't enough time for the majority of students to finish lab 2 last week, and those students didn't get to try the many-to-many task in the lab time. I went over this taks in the lab this week, though. The six week module format doesn't give a lot of time for labs, especially as the first week was before the lecture for some groups, and the last week will be given to working on the coursework.

Posted by Jim Paterson at [04:05](#) [0 comments](#) 

Labels: [class test](#), [peerwise](#), [sql](#)

Tuesday, 10 November 2009

[Reflections on week 2](#)


It was a bit of a rush to get through all the relationships material by the end of the Monday lecture, but this is really needed for the labs this week. The Monday morning lab group were already struggling with task 3 of lab2 as they were doing the activity before the lecture. It would have been very useful to to a live demo of creating tables and creating related tables with SQL rather than just showing the SQL on slides - I really should do this for next year, although it might make it more difficult to get through the material. One problem with a short module is that you need to hit the ground running and get through enough material early on to be able to do meaningful labs and tutorials. I should think about other places in the lectures where live demos can be used to make the material more digestible.

One question which the week 2 activities highlight is whether Access is the best choice of DBMS for this module. Problems with Access include non-standard SQL, particularly datatypes, which can cause confusion, different names for the same datatype in Design View and in SQL, lack of ability to set precision of numbers and to define constraints in SQL View (leading to the need for 'tweaks' in table design view after creation) and quirky behaviour when working with INSERT queries. Students usually do get the hang of these, but I need to look for ways of smoothing the path here (list SQL types in both tutorial 2 and lab 2?). Access does have some major advantages, though, including general availability in the

university and at home, lack of setup required to give students access, the single-file database format for easy portability, inclusion of form and report tools within the same environment and the visibility of the database objects when working. I'd be interested in alternatives, though, and in thinking about how the module would look using Oracle, for example. This module is intended to give an introduction to the idea of SQL and its capabilities, and it is not intended to give comprehensive coverage of SQL (they get that later with an Oracle-based module), so I still think it's OK here.

Some computer labs are better than others for lab 2, and it is difficult to use a consistent approach for all groups when they are in different labs. Some labs don't have projectors, most don't have a separate instructor PC for the projector (meaning that with a large group someone is inevitably using the projector PC), and so on.

It would be nice to be able to do lab 2 after tutorial 3 (summary schemas) and work from a schema design rather than straight from the data model. This would be difficult in the six-week format, but it's worth thinking about for next time. One solution would be to give partial solution to lab 2 after week 2, and to allow students to continue it in week 4, and do lab 4 if finished.

Posted by Jim Paterson at [00:56](#) [0 comments](#) 

Labels: [Access](#), [reflection](#)

[Reflections on week1 \(of 6\)](#)


Nearly at the end of the week, and it's been a hectic week. There are seven tutorial/lab groups this year, and I'm doing all the tutorials and five of the labs. I'm not sure who gets the best deal out of the tutorial - not the first group as it's my first time through for the year, and not the last group as I'm on auto-pilot by then! Lab rooms seem fine this year, but some of the tutorial rooms I've been timetabled in cause problems, especially the room with the "whitewalls" - no whiteboard as such, just white sections on the walls which people use, but where writing can't be rubbed off if it's left for more than a few minutes. I managed to get moved from that one for one class - to a room where the "fixed" whiteboard was propped up against the wall behind a set of partitions. These rooms all have projectors now, which may be handy later, but this "data model design" exercise really needs me to write and draw ideas as they come from the students.

The lab this week is a gentle intro to Access, and runs fairly smoothly. About 2/3 of students have used Access before, very few have used SQL - about the usual. I try to make improvements/corrections to lab sheets right away when I become aware of the need, as I'll never remember later, and put the revised version on the website. I added a note in the lab sheet about enabling content in Access, which comes up quite often in the lab. Introduced a Reflection on lab exercise on Blackboard (write three things you learned in this lab), initially to give a bit of extra work to do for people who finished too quickly - it may also give some useful information and be a useful process for students. Relatively few students have completed this, 28 so far with only one group still to do the lab. One student stated that he already knew everything in the lab...

I changed my approach to the tutorial slightly as I went through the week. It's basically a revision of UML modelling which they've been doing in another module in the previous six weeks, and they should know about use case and class diagrams. I then talk about data being stored or retrieved in the implementation of the use cases, and data being stored

corresponding to classes in the class model, which I talk about as the data model. With the first groups I asked what a use case diagram is, and what a class diagram is, and got very little response. With others, I changed to drawing a simple use case and asking "what is this?" and got a good response. Using similar tactics showed that they could also recognise classes, attributes and relationships (with multiplicities) when I drew a class diagram. We then go on to think about entities for a car hire system. A couple of new ideas came up here which previous years hadn't thought of. Service history is a good idea for an entity with a one-to-one relationship with Vehicle. Somebody suggested "Rental history" for a customer - this gave an opportunity to show that a Hire entity gives a way of storing this information. We had some good discussion about justifications for new entities - groups of related data (category, rental cost, etc) giving a Category entity, repeated data in many Car instances (model, etc) giving a Model category, need to store information about an association (Customer-Hire) giving a Hire entity.

Lectures - got up to the start of relationships and foreign keys, cover this topic on Monday. I do not like two-hour lecture slots, as I have on a Friday. I have an "extra" one hour on Monday, but I've been "persuaded" not to use this every week as one group has no other classes that day... On past experience I should be able to get through the material, though.

Posted by Jim Paterson at [03:24](#) [0 comments](#) 

Labels: [reflection](#)