INTRODUCTION

A database is a collection of information that is organised in a structured way which can be assessed easily by the users. It should contain as little redundancy as possible. It allows users to make changes to the database such as updating the information. It also allows the users to view the information they wish to see. It may also allow multiple users to assess the database. It is useful in handling large amounts of information effectively so that the user can search for or sort information with ease which would be difficult to do manually.

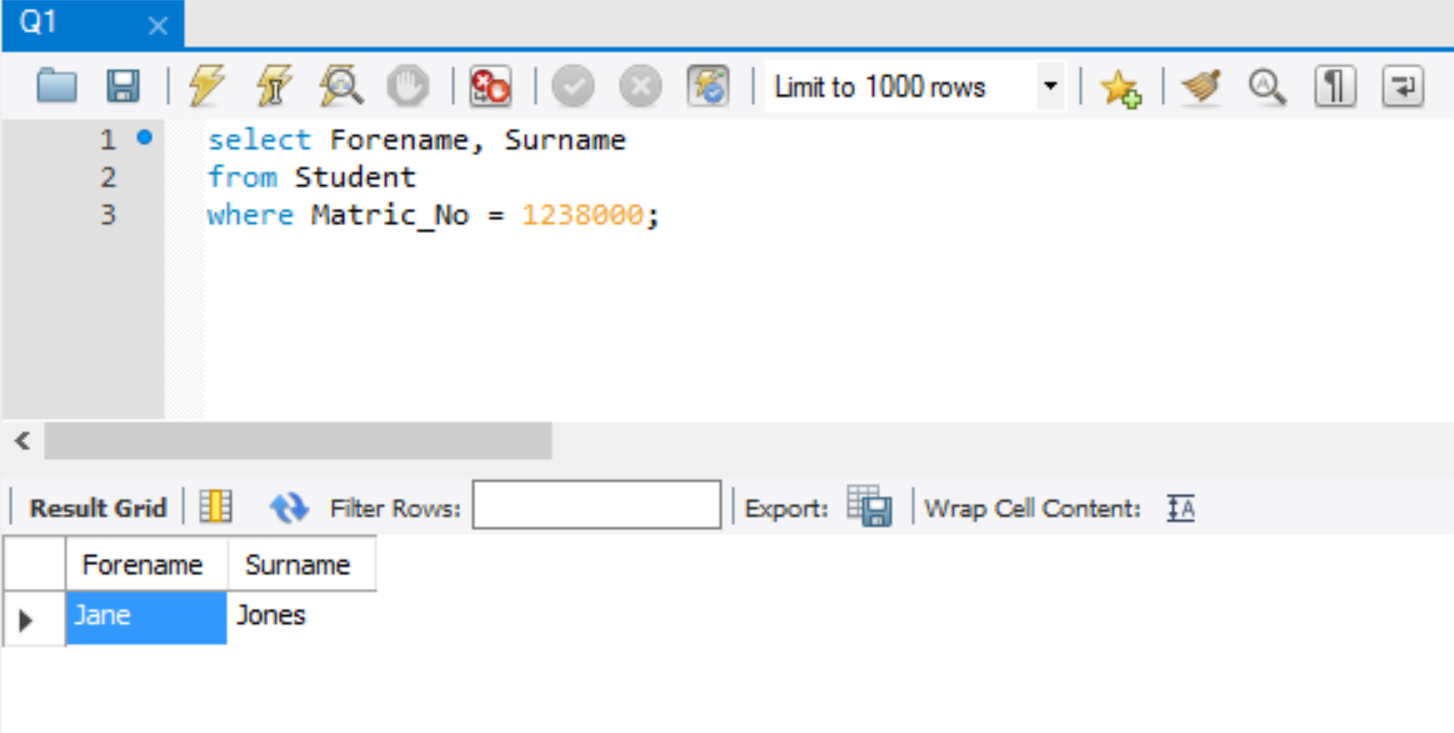
The database I have built contains information for the school of computing science. Which contains details of its courses, students, tutorial groups, staff members and their relations. The database built can be used to return information of many sorts.

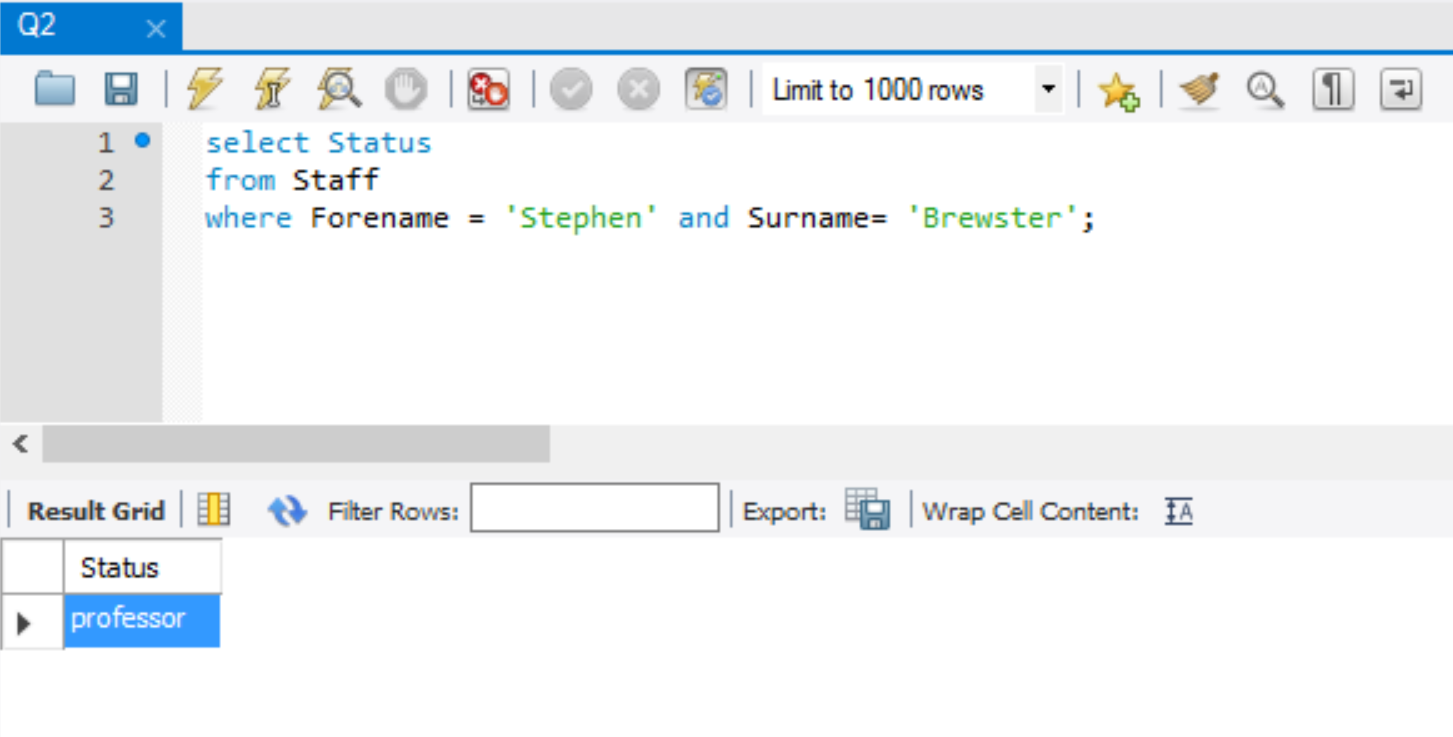
This report will give a short descriptive account of the different step taken to complete the database, which are: Conceptual design, relational schema design, database implementation, populating the database and querying.

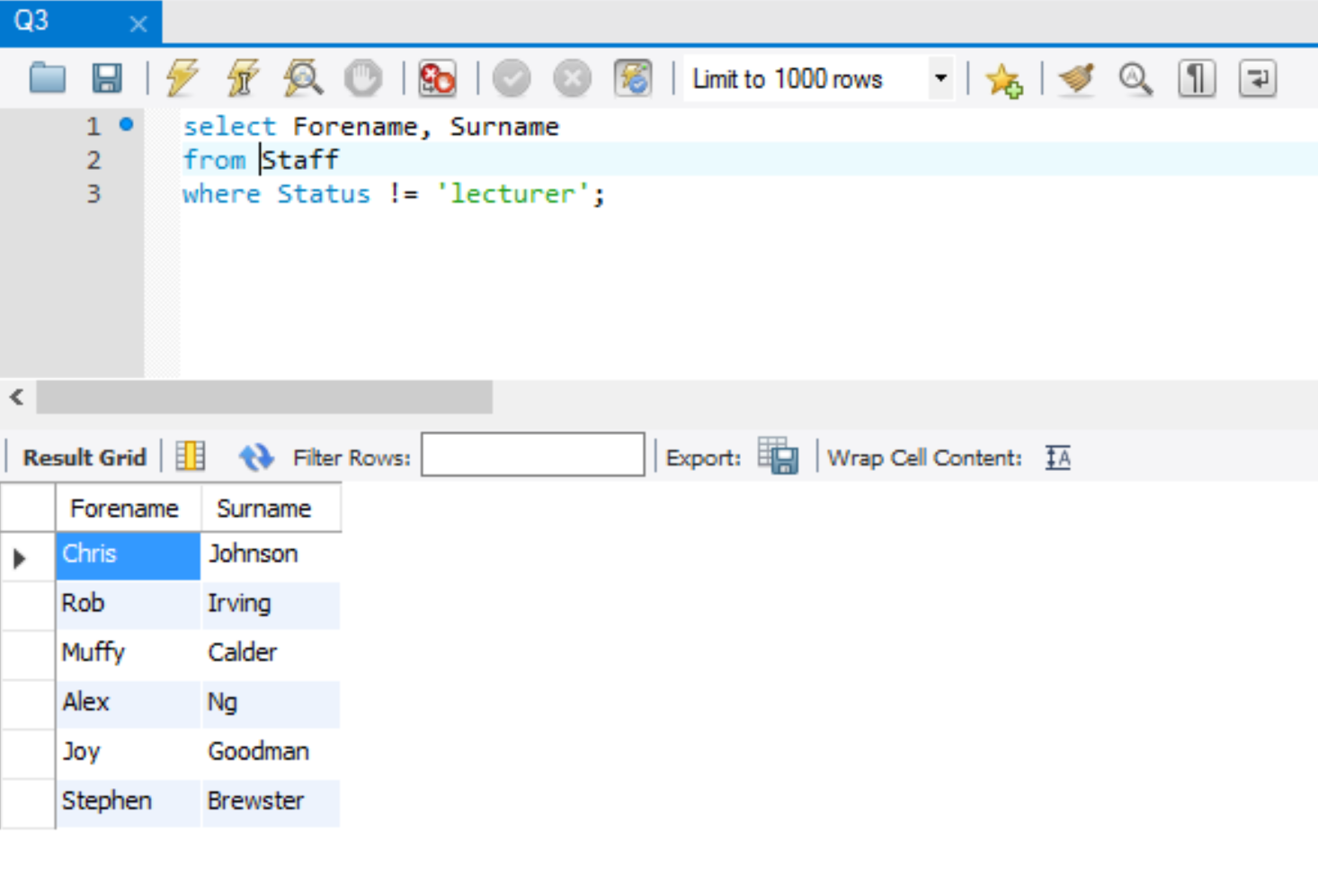
Queries & Results

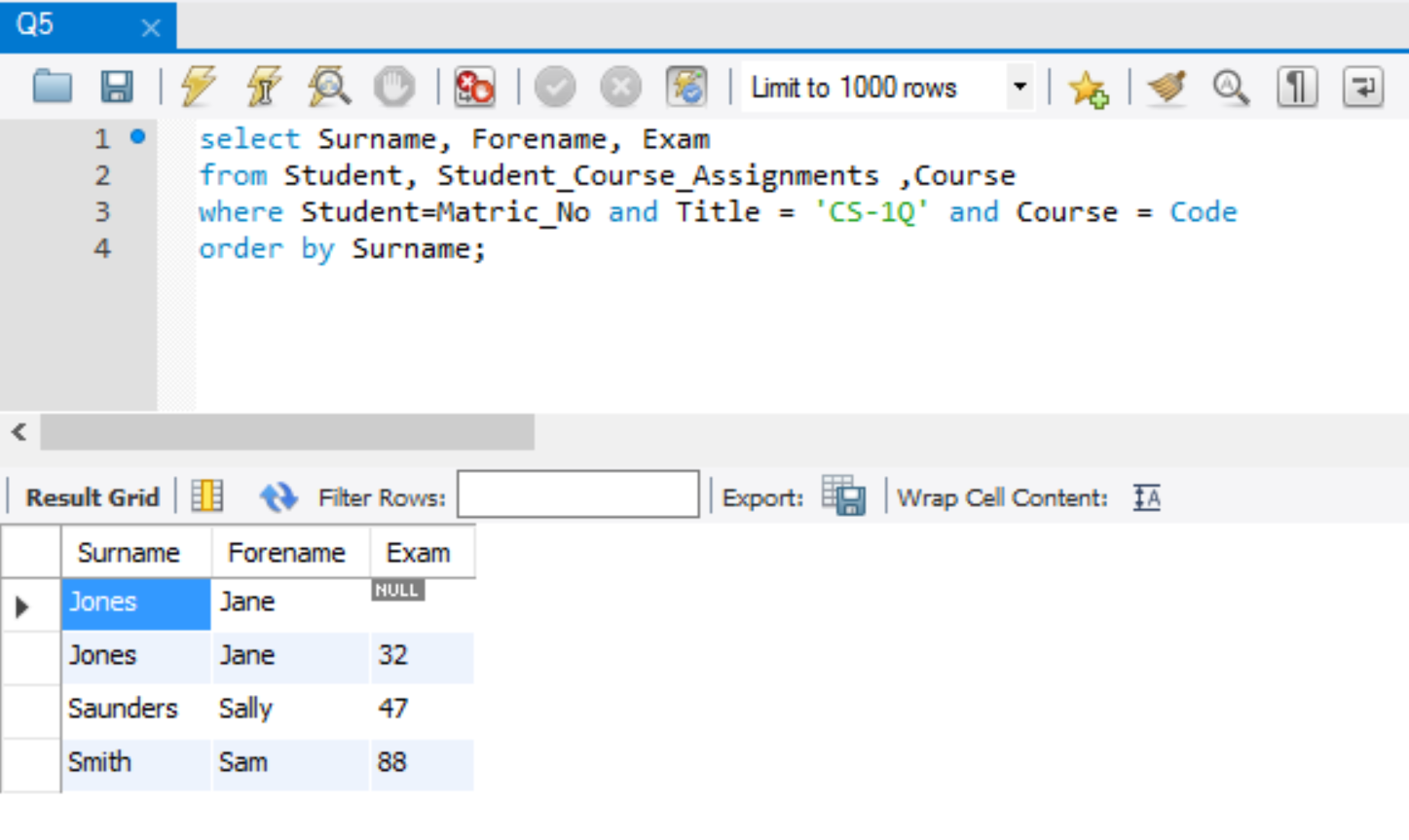
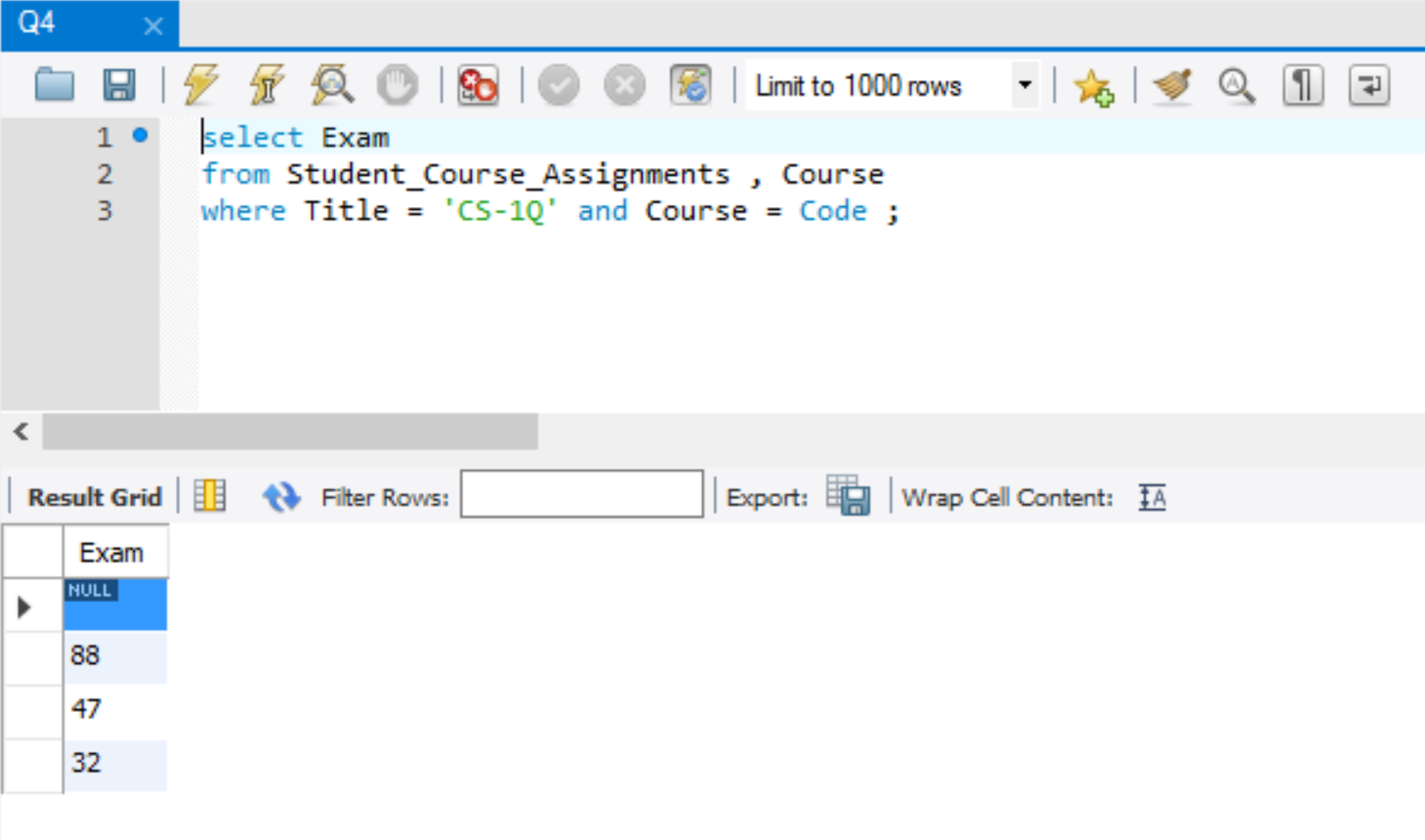
Below are the SQL and results for the queries 1-6, taken as screenshots from MySQL workbench.

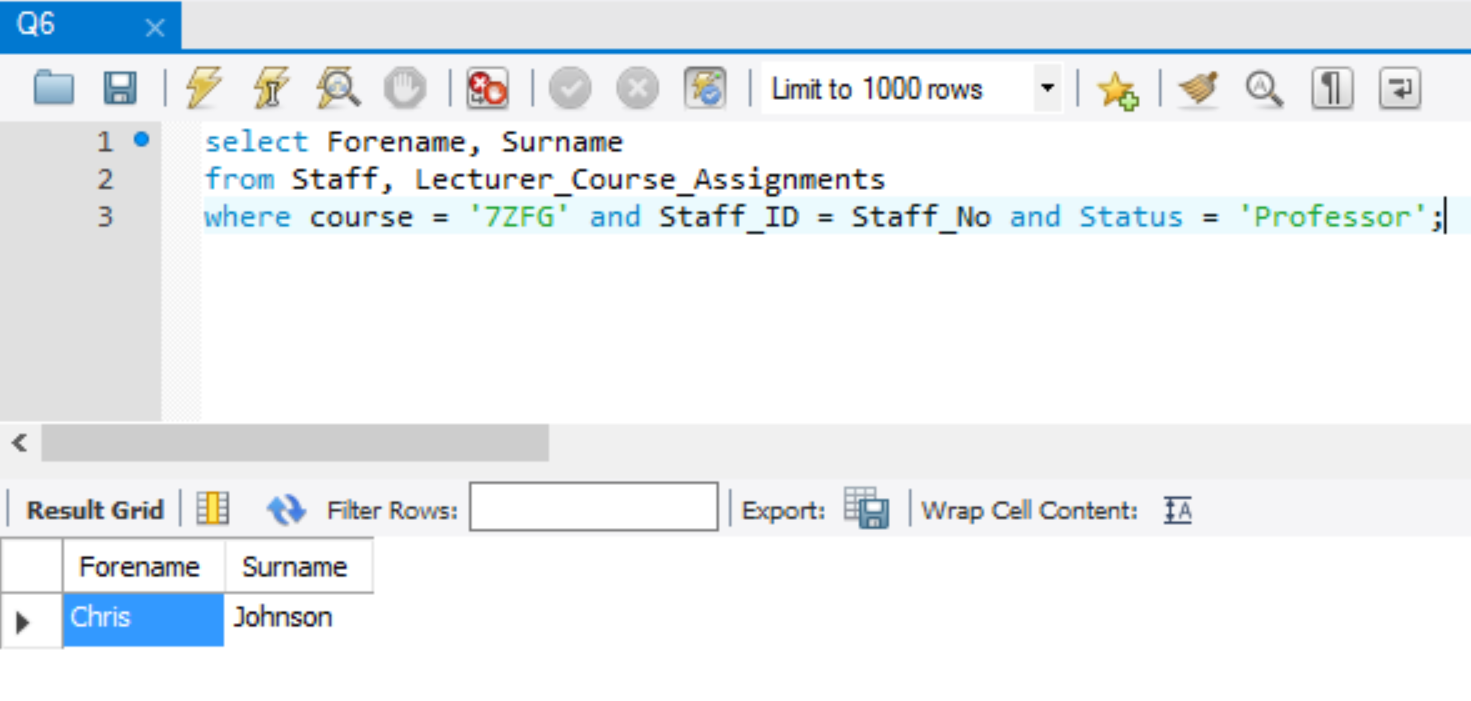
The default schema is set as 2258561l.











Description

Firstly, an Entity-Relationship Diagram is formed consisting of the different types of information to be stored in the database, which are: entities, their corresponding attributes and relations between entity types. The entity types such as strong and weak entity types are identified and some attributes are chosen to be the primary keys. The cardinality and the participation of the relationship are then decided.

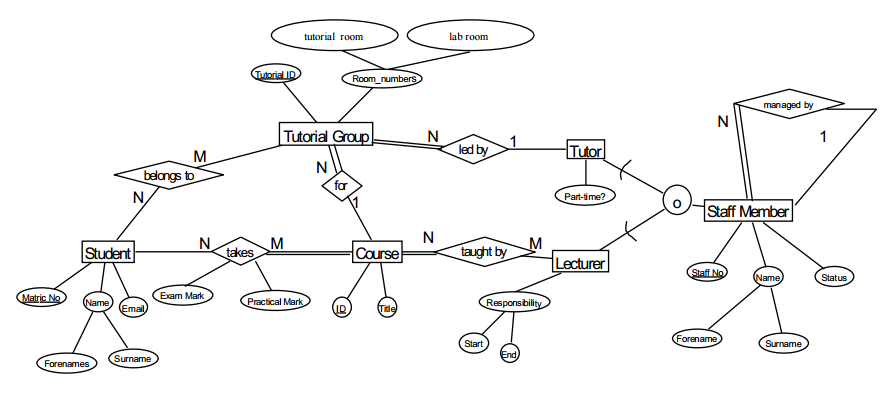
For the relational schema design, the conceptual ER model is turned into a table. The default schema is set. Each entity is a table/relation. For each table, the relation variable is the table name, which is the name of the entity. The heading consists of the attributes of the entity, arranged on the top row, it is also known as the degree of the relation. The body of the table are the data. Then, the foreign keys are identified. A foreign key is a primary key for one table but exists in more than one table, it is used to cross-reference tables and connect the data of different tables. Sometime, with large amounts of data, it is more rational to import data from given files.

Next up is the database implementation, which is how the table was built in MySQL. Data is arranged logically in tables. Each entity is a table. Strong entities are built first with their attribute names, followed by weak entities. After that, foreign keys are selected. For each table, there must be at least one primary key to uniquely identify each table. For tables with foreign keys, the foreign keys must be set as the primary key first before setting it as the foreign key. Data can be manually typed into the tables one by one by hand, or it can be imported into the tables from CSV (common separated values) files. or this database, the provided data are imported into the tables.

After database implementation, the tables are populated. Populating the database can be done by typing in manually or by importing in from files. When importing data, make sure that the data types match and that the data in file matches the column headings in the tables. When importing ‘NULL’ value as a string, remember to set it to NULL afterwards. It is crucial to populate the data for the strong entities first before populating tables with weak entities. Adding data for foreign keys in a table without adding data for the table containing the primary key which the foreign keys are from will result in this foreign key constraint error(error 1452). The value inserted in a foreign key column must already exist in the referenced primary key. Therefore, the order in which the data is entered in the tables is very important.

Finally, queries are performed. Queries are used when we wish to select certain group of information from the database. It is done by combining relations from different relations and tables in a way that gives the desired information. Relational algebra plays an important role in querying in MySQL, it is a set of operations which makes it possible to obtain the desired results of a query. The principle relational operations are select, project and join. Select pick rows from a relation by some condition. Project pick columns from a relation by name. Join connect two relations together, usually by a Foreign Key. The conditions are set operations. Some set operations are: Union, intersection, difference and Cartesian Product. For the query performed for this exercise, which are queries 1-6, only a few simple conditions were used. For each query, there are three basic key words: ‘select’, ‘from’ and ‘where’. The select statement is used to return data from tables. The from clause is used to list the tables and any joins required for the query. For the where clause, different conditions and comparison operators are used to filter the results to be returned. For Query 5, ‘order by’ is used to order the surnames in alphabetical order.

The final choice of ER diagram used for the implementation of MySQL is taken from the model solution form tutorial one. The original ER diagram made was not suitable for the implementation due to missing information and improper relations identified. The final choice of the ER diagram is as shown below:



The major problem for carrying out the work is working on MySQL due to unfamiliarity of the software and how to operate it properly. Trouble shooting was very time consuming without the help of the tutor. It involved lots of Googling, changing many different names for the foreign keys and deleting and rebuilding some tables again. However, it was overcame by asking tutor for help and by trying out different ways to solve the problem.

Conclusion & Reflections

Personal achievement and difficulties.

I now have a more in-depth understanding on how to operate MySQL and am more skilled in querying after trying it out hands on. I have learnt how to properly design and implement a database from start to end.

However, the ER diagram I’ve designed was different from the solution diagram. I should be able to design a better ER diagram. Also I should be able to identify the different relationships better.