

## CSC2041/42 Information Management/Modelling

October 16<sup>th</sup>, 2018

### Group Assignment

---

This will be a group assignment. Each group will work as a team to produce one final result for the project. Co-operation among members within a group is strongly encouraged. Each group should work independently, and no collaboration is allowed across groups.

**Submission Deadline:** Thursday November 29<sup>th</sup>, 2018 by 12 noon. There will be no extensions. There will be a separate email by Week 8 on how/where to submit.

**Group Formation:** Groups have been formed randomly using a computer program, and have been published in a document group-allocations.pdf in QOL resources. Some details of group allocations are also included in that document.

**Problem Statement:** Queen's Accommodation have approached you to help build a database management system that will enable them to keep track of their properties, tenants and employees. The following is the context statement that they have provided:

---

Queen's Accommodation (QA) enables new employees joining Queen's University Belfast to settle in quickly by providing them high-quality accommodation in the vicinity of the university. QA also arranges for accommodation for guest visitors to the university who come for short periods of stay at Belfast. QA owns several buildings, each of which have a distinct address including post code. There are several apartments within each of those buildings. Each apartment in a given building is assigned an apartment number that is unique within that building (but not necessarily across different buildings). QA wants to keep track of the number of bedrooms, the number of bathrooms and the total area (in square feet) for each apartment. The system manages information for different kinds of people. As of now, QA manages two disjoint categories of people: tenants and employees. For each person, the system would maintain emergency contact information, specifically, the name and contact number of a person to be contacted. Each person is identified by a unique ID, and has information consisting of their first name and last name.

For each tenant, it is necessary to record their bank account number, since rent needs to be collected every month. For each employee, the application will keep track of their monthly salary. Employees may be managers or technicians (or both). Each manager has an office located in one of the apartments owned by QA. Each manager manages at least one apartment, and each apartment must have a manager. For each technician, the system would keep track of the skills they have; which may be one or more of the following: "carpentry", "plumbing", "electrical".

To rent an apartment from QA, tenants must sign a lease agreement. A lease agreement is a formal contract, identified uniquely by an ID, signed by one or more tenants (as may be the case with roommates) to lease an apartment. It includes the start date and expected duration of occupation (in terms of number of months), as well as the monthly rent charged for the apartment. The lease agreement is also signed by any one of the managers at QA. The system keeps track of all (expired and live) lease agreements signed so far. Similarly, even if a tenant ceases to be one, her details are not purged from the system in anticipation of future custom.

---

This assignment has four tasks:

- (i) **E-R Modeling:** Generate an E-R diagram for the above database project for Queen's Accommodation.

The output of this stage will be an E-R diagram (1 page) and a list of constraints or assumptions (if any) that you have made in the course of designing the E-R diagram (1 page max). Feel free to make reasonable and intuitive assumptions, but do list them.

- (ii) **Database Design:** Design the relational database for the task. You need to decide things such as the attributes in each table, the primary key and foreign keys, and make design decisions such as whether there are any redundant tables that can be reduced into other tables.

The output of this stage will be in the form of table schemas (relational schemas). An example relational schema for a simple table would be "table1(attr1, attr2, attr3)" that indicates that the table contains three attributes, with two of them (the underlined ones) forming the primary key. For each table, you would also include information mentioning the entity sets and relationship sets you have modelled in the table, foreign keys, and optionally, any remarks that would help convey your thought process. This can occupy up to 1 page in your report. Additionally, create two files:

- createdbx.sql: The script for creating the tables where  $x$  should be replaced by your group number (e.g., createdb6.sql for Group 6 and createdb56.sql for Group 56)
- cleandbx.sql: The script for cleaning up (i.e., dropping) all the tables you have created. Once again,  $x$  refers to the group number.

Note: Be aware of foreign key constraints while writing up cleandb.sql, and ensure you drop the tables in a valid order. If your script does not execute well (e.g., throws an error due to foreign key constraints), marks would be deducted.

- (iii) **SQL Querying:** Think of 4 real-world and reasonably complex scenarios of querying the database (e.g., find those tenants who have stayed in two different buildings within the same year [note: do not use exactly this example query]), and design SQL queries for those scenarios. For each scenario, give a verbal description of the query, and provide an SQL statement for the query. You are also encouraged to give verbal descriptions to aid understanding of the SQL query itself; e.g., for SQL statements with multiple nested queries, you may include verbal descriptions of what each of the nested queries do. Each scenario along with the SQL query should be described completely in one page (note that the SQL query with good indentation can itself take a significant amount of page space). Thus, the response for this section would be up to 4 full pages. Additionally, the following files need to be provided:

- insertdbx.sql: This script would insert sample data into the tables created in createdb.sql
- querydbx.sql: This script would contain the 4 SQL queries. It is expected that the insertdbx.sql is rich enough that each of the queries would give some tuples in the result.

Note: Do ensure that you enter enough data into your tables through insertdb.sql so that your queries return non-null result sets even if they are fairly complex and involve many joins.

- (iv) **Coping with Changes:** Queen's Accommodation would like to evolve as time goes by, and significantly expand its services. For example, QA may find that it is running short of apartments to fulfil the requests it gets, and decide to expand by acquiring apartments on long lease from private owners in Belfast, and offer them to their clientele (i.e., QUB visitors and staff). Think of a reasonably sophisticated scenario like the above (note: please do not use exactly the above scenario) and produce a report on how you would address such an expansion plan and detail the changes you may need to make to your system. You only need to verbally describe the changes, and the design choices you may have to make in order to

support the expansion plan you have chosen to address. Think of this as a report to a database expert detailing the nature and complexity of the effort needed to address the particular QA expansion plan you have chosen. Be concise and to the point. The full report for this task, comprising the description of the chosen expansion scenario and the means to address it, should fit in 1 page.

- (v) **Question:** Which was the one entity set or relationship set from the problem description that you found hardest to model in the E-R Modeling task. Why do you think you found that hard to model? [This task is **unassessed/unmarked**.]

**Assignment Material:** As indicated over the various sections above, the final output from the assignment would be the following:

- Report named reportx.pdf (*x* should be replaced by the group number) structured as the following:
  - Page 1: Cover Page, with information of the group members (student numbers, as well as e-mails)
  - Page 2: E-R Diagram (Ref: Task (i))
  - Page 3: List of Constraints and/or Assumptions in E-R Diagram Design (Ref: Task (i))
  - Page 4: Database Design (Ref: Task (ii))
  - Pages 5, 6, 7 and 8: SQL Querying (Ref: Task (iii)) – each one should be 1 page max.)
  - Page 9: Coping with Changes (Ref: Task (iv))
  - Page 10: Answer to hardness question (Ref: Task (v))
  - Page 11: Individual Contributions Record [Refer below]

Note: These task-wise sub-limits on pages need to be respected. Any violations on report format or section limits would be penalized by up to 10% of total allocated marks.

- Set of Scripts: 4 files named as the following (where *x* is the group number):
  - createdbx.sql
  - cleandbx.sql
  - insertdbx.sql
  - querydbx.sql

**Individual Contributions Record:** The final page in the project report, as indicated above, would be an individual contributions record, which would be in the form of a table as below.

Group Member St. Number	Task (i)	Task (ii)	Task (iii)	Task (iv)	Group Member Totals
1234567	10	10	10	50	<b>80</b>
4531245	15	...	...	...	...
...	20	...	...	...	...
...	5	...	...	...	...
...	40	...	...	...	...
...	10	...	...	...	...
	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	

While most of the table above is self-explanatory, the main information is noted below. There would be one row in the table for each group member, who is identified by the student number. Each of the four tasks are identified by a column. Each task has 100 contribution points, which need to be divided

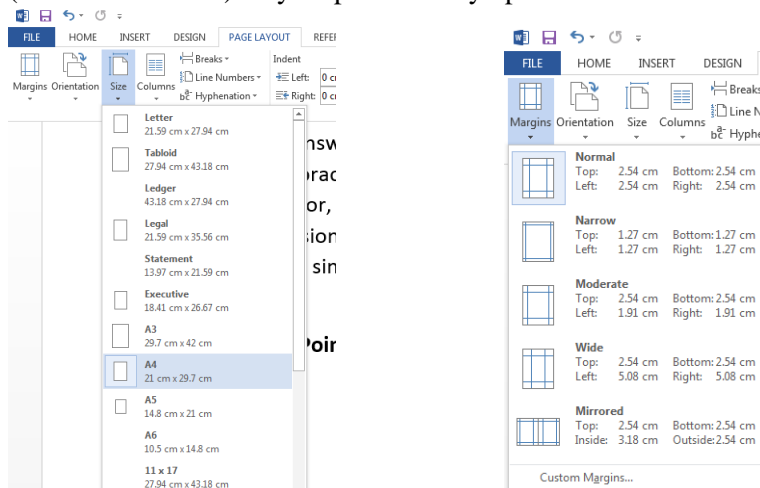
across the members of the group; the numbers in each column should total 100. The total points in each row are recorded separately in the column titled Group Member Totals.

**Assessment:** The first four sections carry marks in the proportion 3:3:3:1; thus, the ER model accounts for 30% of marks, whereas the coping with changes accounts for 10%. The fifth section is not marked. There will be one final score for each group. The individual marks will be arrived at by multiplying the final score for the group by a contribution factor. The contribution factor would be determined using the Individual Contributions Record, and will be directly related to individual contributions reported therein.

**Support:** The demonstrators assigned to the module will be available to help with the project to answer specific queries (e.g., requests for clarifications on the problem statement etc.) during the practical sessions or by email at [csc2041@qub.ac.uk](mailto:csc2041@qub.ac.uk) or [csc2042@qub.ac.uk](mailto:csc2042@qub.ac.uk); these mailboxes are monitored by demonstrators, who will respond to your email. Please note that questions on design decisions (e.g., do you think X can be made a separate entity set/table?) will not be encouraged since that would give be disadvantageous to the groups who decide to figure it out themselves. The list of demonstrators and contact information may be found in the demonstrators.pdf document on QOL.

### Important Points to Note:

- **Report Formatting:** Please use the A4 page size (choosing it in Microsoft Word is shown in the screenshot below), and use the Times New Roman font with a minimum font size of 11pt (larger font sizes are fine, but they reduce the amount of text you can fit within your page limits). Additionally, you should use at least 1-inch (2.54cm) margins on all four sides, which maps to the “Normal” page margin setting in Microsoft Word (screenshot below). Any violations of recommended report formatting or non-adherence to task-wise page limits (mentioned earlier) may be penalized by up to 10% of total allocated marks.



- **File Naming:** As mentioned already, all files (reports and scripts) need to be named with the group number as the final part of the name. For example, Group 9 would name its report as report9.pdf. There should be no leading 0s; i.e., do not use filenames such as report09.pdf.
- **Submission:** The final submission should be in the form of a zip file comprising the 5 files described in the “Assignment Material” section above. The zip file should be named using the above convention as assignmentx.zip (e.g., assignment9.zip). You will be provided directions as to how to submit the assignment, by Week 8.
- **ER Diagram:** For the ER diagram, follow the formatting conventions that have been discussed in the lectures and included in the slides. Different textbooks use very different

conventions, so be careful of making sure that you adhere to the formatting conventions that we have discussed.

- **Clarity:** Since we will not have an oral presentation, the report forms the primary means of evaluation. Thus, the onus is on you to ensure that there is no ambiguity in the report (for example, name attributes well). Lack of clarity that would allow for misinterpretation could mislead the evaluator and thus lead to loss of marks.
- **Updates to this Assignment Document:** This document may be updated in course of time to include additional information (we do not expect to make changes or remove content from the document); so, please check on QOL for updates to the document. Updates will be numbered separately, and older versions will not be deleted.