Glasgow Caledonian University

SESSION: 2021/22 DIET 1

Module Title: Database Development

Module Code: M1I325894
Level: 1

Module Leader: Dr Ryan Gibson

Coursework Description

Contents

Scenario and ERD

An academic institution has previously relied on a system of spreadsheets to represent various entities of interest to the organisation. However, as a small trial they require you to develop a supporting database, which will be used to query certain subsets of data in a more efficient and robust manner. To model this scenario, a number of tables are required. These tables are presented in Figure 1.

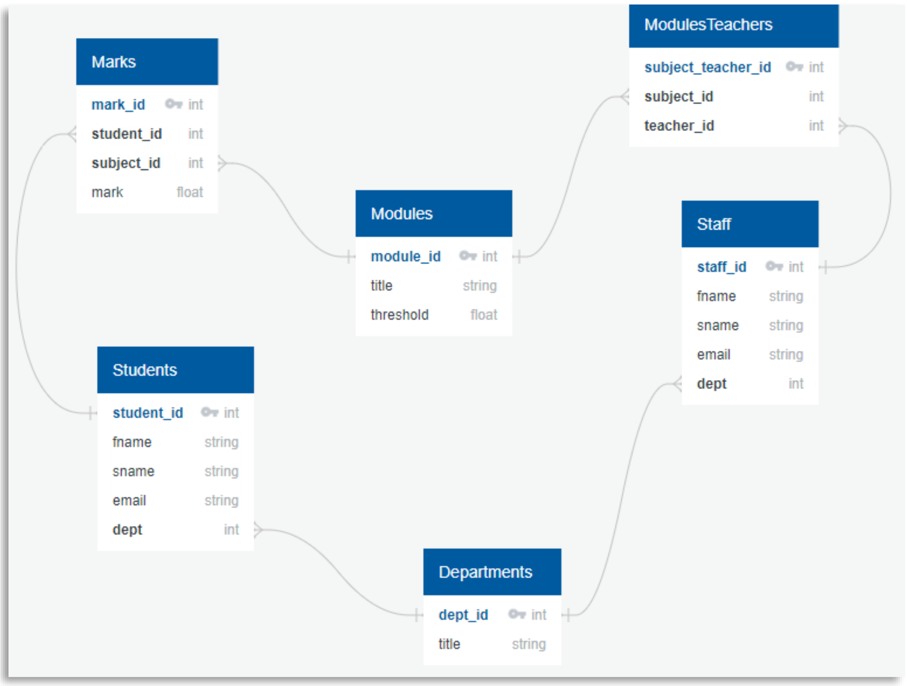


Figure 1: Initial Entity Relationship Diagram

Coursework workflow

A folder named <*surname_forename.zip*> is available. Download this and extract it to your machine. Rename the folder appropriately, i.e., if your name is John Travolta, then name it *Travolta_John*.

This folder, named after you, will be your *working directory* while you are developing your coursework. Note: the folder contains a number of <*.sql> files, <*.docx> files and <.csv> files. Don't let the number of files be of concern; as you read and work through the problem, you will realise that each file only requires a relatively small amount of work, and you will recognise the motivation for this from the work you have completed in the labs.

Please read the coursework task carefully, and good luck.

Coursework tasks

Suggested, preliminary task [no marks]

Note well: the provided scripts <.sql> should be used to place your SQL code. **Do not** rename these files. The top-level file <*top_level.sql*> has been completed for you. It might be an idea to take a look at this in notepad++, and call it (using sqlite3) to see the output, just as a starter, i.e.: if calling a local copy of sqlite3 in PowerShell type:

```
./sqlite3.exe university.db
```

which will load the existing empty database named <university.db> into memory. Then, remembering to use the ‘dot’ before ‘read’ type:

```
.read top_level.sql
```

This script has been written for you; take a look at the content and observe the output from that script, which is derived from the ‘echo’ calls, and you should understand the call hierarchy. Therefore, when you see such that when you have successfully completed the other scripts, calling top_level.sql *should* create and load the whole database.

Task 1 [total marks = 52]

Task 1a: SQLite3 tables and data loading [**30 marks**]
The task is to create an SQLite3 implementation of the ERD presented in Figure 1.
For **each table in the ERD** (there are 6), complete the two appropriately named SQL scripts:

- create_*.sql
  - This script will implement the respective schema.
  - The script should contain logic that removes a table, before recreating it, if it exists.
- load_*_data.sql
  - This file will load the data, which is provided in respective .csv files.
  - Notice the .csv files, one corresponding to each table. You can use the data in these files to load into your database, once you have implemented the design. It is up to you how you do this, either using pure SQL or utilities available in SQLite3.

5 marks for each table for successful *creation* and *loading* = [**30 marks]**

A top-level <*top_level.sql*> script (created for you…you do not need to edit this script) calls the *create_*.sql* files and the *load_*data.sql* files. Therefore, on successful completion of this part of the coursework, a database is implemented, and data is loaded into it on calling *top_level.sql*. The following task (Task 1b) will query the database and print output to the console.

Task 1b: *report.sql* script [**22 marks**]
The *top_level.sql* (right near the bottom) calls *report.sql* script. In this task, you are required to write a set of SQLite3 queries in the *report.sql* script, which will:

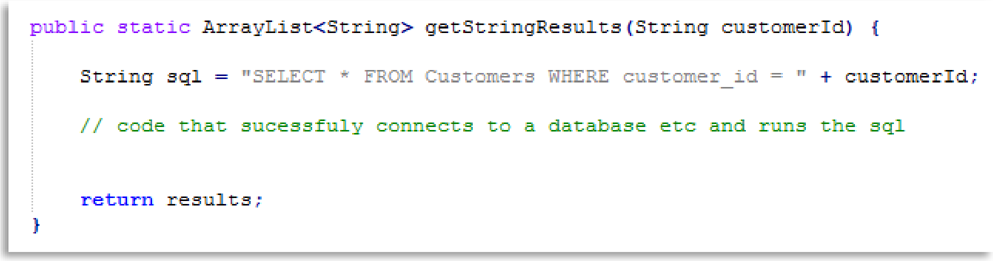
1. List all columns from Departments, Modules, Staff and ModulesTeachers [**2 marks**]
2. List the average mark for all marks. [**2 marks**]
3. List the first name, surname and email of staff, ordered by e-mail, but only list 10 of the staff. [**4 marks**]
4. List the forename and surname of the students and their marks per module. [**8 marks**]
5. List the title of each module, and the average mark obtained for that module, and order the output from highest to lowest average mark. [**6 marks**]

Task 2 [total marks = 48]

Inside the document named <**answers.docx**> there are 3 questions:

1. *Please explain what foreign keys are and why they are used in a database, using examples from the coursework.* [**16 Marks**]

2. *How is a method, such as the Java method below, susceptible to SQL injection attacks and what can be done to help prevent them?* [**16 Marks**]



```
public static ArrayList<String> getStringResults(String customerId) {

    String sql = "SELECT * FROM Customers WHERE customer_id = " + customerId;

    // code that sucessfuly connects to a database etc and runs the sql


    return results;
}
```

d

Figure 2: example java method

3. *Why is the scenario, outlined above, a good candidate for a relational database as opposed to a noSQL implementation?* [**16 Marks**]

Complete the document <**answers.docx**>. Each answer should be approximately 200 words in length.

Submission

Process

Before your submission, ensure that you have:

- Followed and completed all the above tasks and;
- Signed the plagiarism statement with your e-mail, see <**cover-page.docx**>

Then zip-up your working directory:

- Do not zip it up into any other format than a <.zip>, such that if I was submitting this coursework the name of my zipped folder would be:
    - <***gibson_ryan.zip***>

…and submit it on the link that will be provided (see Assignments area on GCU Learn).

Deadline

Submit your .zip file before the deadline stated on GCULearn.