INFO20003_2024_SM2

Database Systems
On-campus, Semester 2, 2024

THE UNIVERSITY OF MELBOURNE

INFO20003 Semester 2, 2024

Assignment 2: SQL

Due: Week 8 - Sunday 15th September 2024, 5:59pm Melbourne Time.

Submission - Via LMS https://canvas.lms.unimelb.edu.au/

Case: "Slarc" App

"Slarc": Super Lovely App for Requesting Communications

Description

As fellow Database experts, DOTA2 fans, and enterprising communications fans, you and your classmates have created a new open source version of Teams/Slack called Slarc (inspired by the DOTA2 character and Slack!).

For each user, Slarc records their details such as an ID, a username, an email address, a login mechanism (which is defined strictly as one of the following: Google, Apple, Facebook, GitHub), and a reputation score (which is an integer from 0-100 inclusive, 100 as highly trustworthy and 0 being highly untrustworthy). Users can also upload an image for their avatar.

Users communicate with each other by posting in channels. Each channel has an ID, name, date of creation and optional description. When a user posts in a channel, Slarc tracks the post's author, content and date of creation. Users can also post a reply to an existing post, or react to a post with an emoji, of which the reaction timestamp is recorded. There is an option for users to send attachments in a post; The system records the file size and the dataURL of the object. Slarc automatically scans each post for any harmful content, such as swearing and not safe for work material, and automatically restricts such posts. Similarly, it scans any attached files for potential viruses and flags the results.

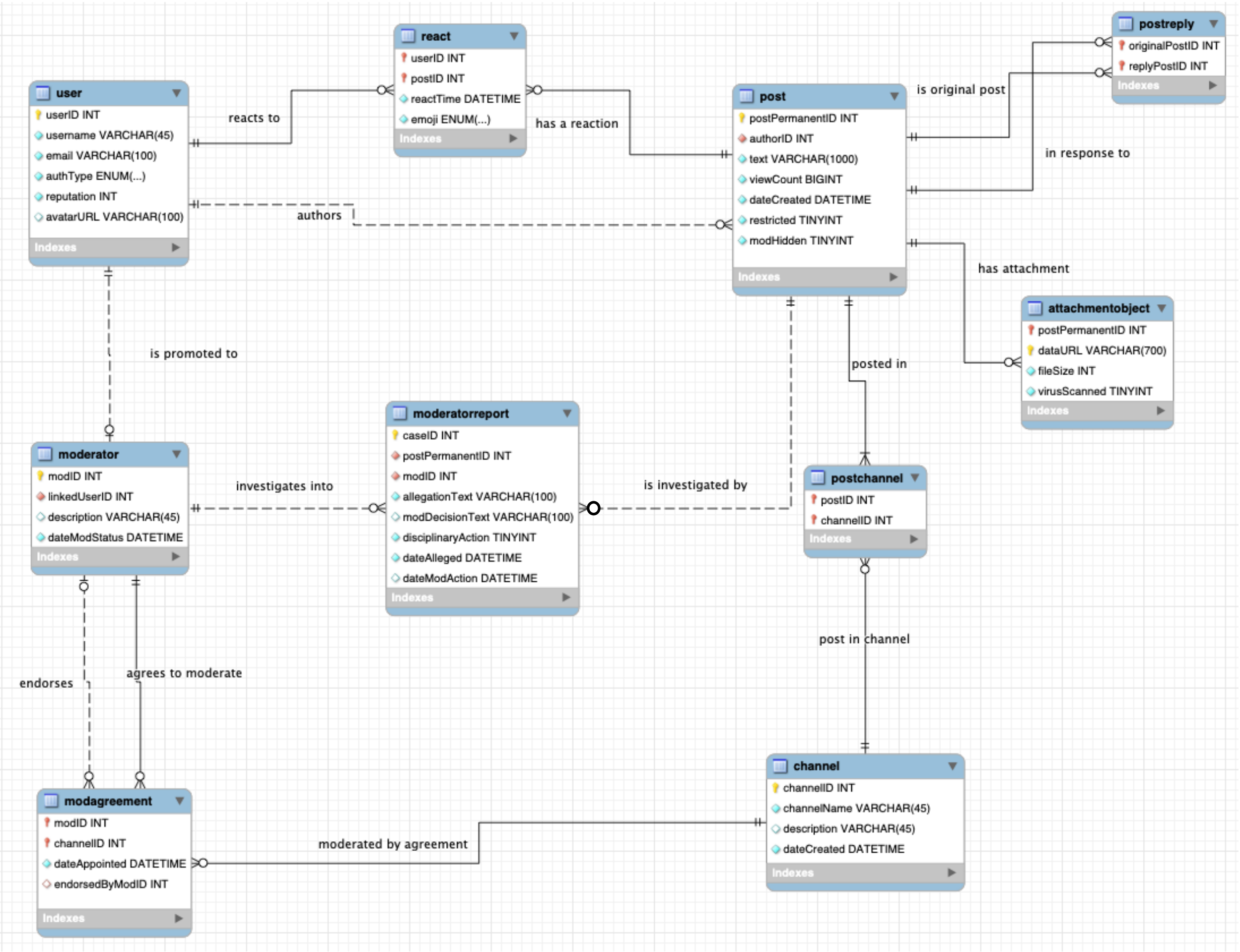
When a user's reputation becomes greater than 80, they can be promoted to become a moderator. Other users will be able to see the date that a moderator was promoted, and their self-description if they choose to write one. Once one becomes a moderator, they can be appointed to moderate channels, usually endorsed by another moderator of the associated channel. Their date of appointment is also stored.

Moderators have the responsibility to ensure that posts in the channel abide by community guidelines. If they come across an inappropriate post or one that is flagged with restricted content, they will need to investigate and report on this case. For each case, the responsible moderator must record the case ID, give a brief explanation of the allegation and decide on whether it requires a consequential disciplinary action. The date of allegation will be automatically recorded. If a moderator decides that there needs to be a consequential action, then they can write down the associated action and its date; they are able to hide any post in the channel.

The Data Model

The Data Model from MySQL Workbench is provided in Figure 1.

FIGURE 1. DATA MODEL FOR SLARC.



Assignment 2 Setup

Please pay special attention to the penalties listed [⚠].

A dataset is provided which you can use when developing your solutions. To set up the dataset, download the file **slarc.sql** from the Assignment link on Canvas and run it in Workbench. This script creates the database tables and populates them with data.

The sample dataset provided is a basic example of Slarc deployed for a DOTA2 e-sports community. You may find that you may need to add some more sample data in Workbench to fully test out each and every query.

Note that this dataset is provided for you to experiment with: but it is NOT the same dataset as what your queries will be tested against (the schema will stay the same, but the data itself may be different). This means when designing your queries, you must consider edge cases even if they are not represented in this particular data set.

The script is designed to run against your account on the Engineering IT server (info20003db.eng.unimelb.edu.au). If you want to install the schema on **your own MySQL Server installation, uncomment the lines at the beginning of the script.**

⚠ **WARNING:** ***Do NOT disable*** **`only_full_group_by`** ***mode when completing this assignment***. *This mode is the default and is turned on in all default installs of MySQL workbench, and we've added a line to the top of slarc.sql to turn it on every time you run the script in case you disable it! You can check whether it is turned on using the command* `**SELECT @@sql_mode**`; *The command should return a string containing* `ONLY_FULL_GROUP_BY` *or* `ANSI`. ***When testing, our test server WILL have this mode turned on, and if your query fails due to this, you will lose marks.***

The SQL Tasks

Please pay special attention to the penalties listed [⚠].

In this section are listed 10 questions for you to answer. Write one (single) SQL statement per question. Each statement must end with a semicolon (;). Subqueries and nesting are allowed within a **single** SQL statement – however, you may be penalised for writing overly complicated SQL statements.

⚠ **WARNING: DO NOT USE VIEWS (or 'WITH' statements/common table expressions) OR VARIABLES to answer questions. Penalties apply.**

? *The Questions*

1. List all posts which contain no ‘react’s. Your query should return results of the form (**postPermanentID, text**). **(1 mark)**

2. Find the most recently promoted mod (moderator) in the entire database. Assume there are no ties (only one is the most recent). Your query should return results of the form (**modID, username, dateModStatus**). **(1 mark)**

3. List all posts created by user ‘axe’ that have at least 9000 views. Your query should return results of the form (**postPermanentID, viewCount**). **(1 mark)**

4. Find the post which is most commented on (Hint: most ‘originalPostID’ appearances). If there are ties, then you must return all posts with the highest number. Your query should return results of the form (**postPermanentID, totalCommentCount**), with one row per post in case of a tie. **(2 marks)**

5. List the dataURLs for all attachments to posts in channelNames containing ‘dota2’ (e.g., ‘dota2_players’, ‘info20003_better_than_dota2’). Your query should return results of the form (**dataURL, channelID**). **(2 marks)**

6. Find which channel has the highest number of ❤s made to posts within the channel. (Hint: ❤s are simply ‘love’ found in the emoji ENUM). If there are ties, then you must return all results. Your query should return results of the form (**channelName, heartCount**), with one row per channel in case of a tie. **(2 marks)**

7. Find the names of controversial users: defined as users who have < 60 reputation, have at least 1 moderatorreport on one of their posts, and at least 3 ‘love’ react’s given to their posts in total. Your query should return results of the form (**userID, reputation, totalModeratorReports, totalLoveReacts**). **(2 marks)**

8. List the top 3 channels with the largest number of posts with attachments identified with virus(es). Also return the total count of such attachments for each channel. Your query should return results of the form (**channelID, channelName, totalVirusInfectedAttachments**). **If there are ties in the top 3 positions, you must return all ties**. For example, let’s say the database contains seven channels and the ‘total number of virus-afflicted posts’ for each channel are (5, 4, 4, 3, 3, 2, 1). The top 3 counts are 5, 4 and 3 so you need to return the top 5 rows, which are the ones having attachment counts of (5, 4, 4, 3, 3). **(3 marks)**

9. We’ll use the term ‘repeater’ to describe a user who has had posts in more than one channel reported. Find moderators, and determine **how many** ‘disciplinary actions’ (as per the **disciplinaryAction** flag) they’ve given to ‘repeater’ users. Your query should return results of the form (**modID, numberOfDisciplinariesToRepeaters**). **(3 marks)**

10. Find users who have *not* posted (or commented on any post) before 01/04/2024 in the channel ranked_grind, but have posted at least one *COMMENT (a post which is a reply to another post)* in dota2_memes on or after 01/04/2024. Your query should return results of the form (**userID**) for all such users. **(3 marks)**

⚠ *SQL Response Formatting Requirements*

Please pay special attention to the penalties listed [⚠].

To help us mark your assignment queries as quickly/accurately as possible, please ensure that:

- Your query returns the projected attributes in the same order as given in the question and does **not** include additional columns.
  - E.g., if the question asks, 'return as (userId, name)', please write `SELECT userId, name` …
  - ⚠ **DO NOT return attributes in the WRONG order,** e.g., `SELECT name, userId`…
  - You can name the columns using `` `AS` `` however you'd like, only the **order** matters. E.g., this is fine: `SELECT userId, name AS fullName`
- Please do NOT use "databaseName.tableName" format.
  - E.g., please write "`SELECT userId FROM users`…"
  - ⚠ **DO NOT provide the database name**, e.g. `SELECT userId FROM coltonc.users` ….
- Ensure that you are using single quotes (') for strings (e.g. …`WHERE name = 'bob'`…) and double quotes (") <u>only</u> for table names (e.g. `SELECT name FROM "some table name with spaces"`…) .
  - ⚠ **Do NOT use double quotes for strings:** …`WHERE name = "bob"`….
  - ⚠ **Do NOT use Microsoft Word 'smart quotes'** (the fancy ones as you see in "this" 'example').
- Comments are optional, but we recommend writing them for complex queries.
- ⚠**Do NOT delete the special comment markers in the SQL template file.** These include: `-- BEGIN QX`, `-- END QX`, and `-- END OF ASSIGNMENT` (where X is the question number). They help us mark your submission so tampering with them will hinder our marking and **will** attract penalties.

Assignment Submission Instructions

Please pay special attention to the penalties listed [⚠].

Your submission will be in the form of an SQL script. There is a template file on the LMS, into which you will paste your solutions and fill in your student details (more information below).

This .sql file should be submitted on Canvas by **the time indicated on the first page of these instructions**.

Name your submission as 987654.sql, where 987654 corresponds to YOUR student ID number.

Please make sure that you **actually submit** your file on Canvas ⚠. After uploading the file, you need to press **'Submit Assignment'** to actually submit the file. If you submit late because you failed to press the submit button and only noticed this after the deadline, your submission will be considered late just like any other late submission to maintain fairness for all students.

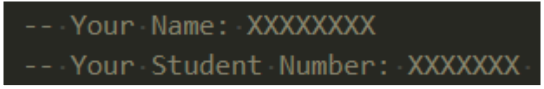
Filling in the template file:

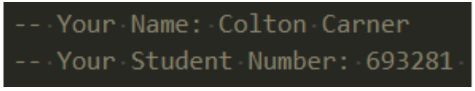
The template file on the LMS has spaces for you to fill in your student details and your answers to the questions. There is also an example prefilled script available on Canvas as well.

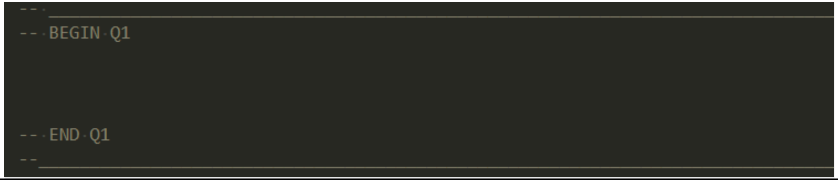
Below (Table 1) are screenshots from those two documents explaining the steps you need to take to submit your solutions:

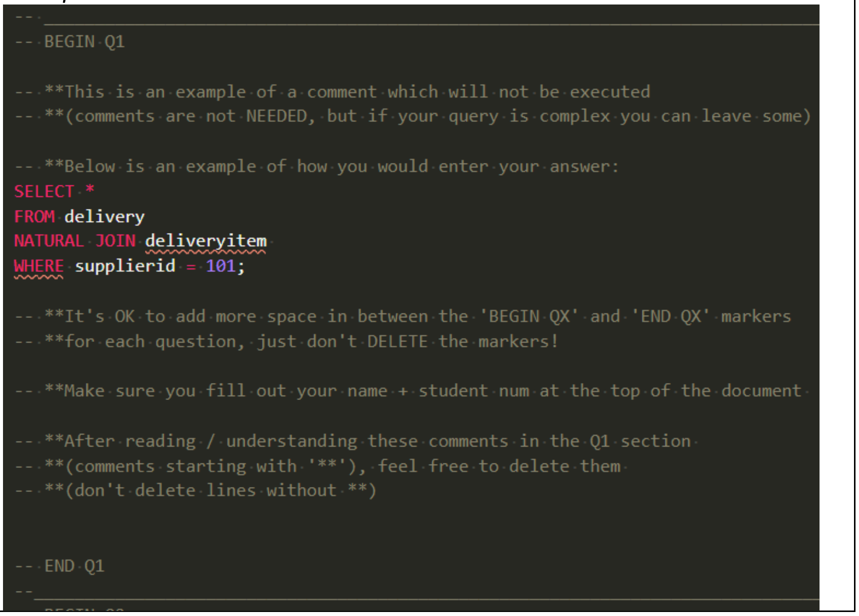
TABLE 1: SCREENSHOT EXAMPLES ON HOW TO SUBMIT THE SOLUTIONS.

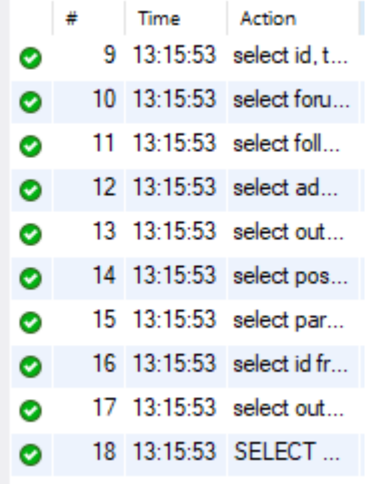
Step	***Example***

1. At the top of the template, you'll need to replace "XXXXXXXX" with your student number and name	*Template*
	
	`-- Your Name: XXXXXXXX`
	`-- Your Student Number: XXXXXXX`

	Example Filled in
	
	`-- Your Name: Colton Carner`
	`-- Your Student Number: 693281`

2. For each question 1-10, place your SQL solution in between the "BEGIN QX" and "END QX" markers.
Ensure each query is terminated with a semicolon ";"
	Template
	
	`--`
	`-- BEGIN Q1`
	`-- END Q1`
	`--`

	Example Filled in
	
	`--`
	`-- BEGIN Q1`
	`-- **This is an example of a comment which will not be executed`
	`-- **(comments are not NEEDED, but if your query is complex you can leave some)`
	`-- **Below is an example of how you would enter your answer:`
	`SELECT *`
	`FROM delivery`
	`NATURAL JOIN deliveryitem`
	`WHERE supplierid = 101;`
	`-- **It's OK to add more space in between the 'BEGIN QX' and 'END QX' markers`
	`-- **for each question, just don't DELETE the markers!`
	`-- **Make sure you fill out your name + student num at the top of the document`
	`-- **After reading / understanding these comments in the Q1 section`
	`-- **(comments starting with '**'), feel free to delete them`
	`-- **(don't delete lines without **)`
	`-- END Q1`
	`--`

3. Test that your script is valid SQL by running it from MySQL Workbench. Run the entire script by copy-pasting this entire file into a new workbench tab, placing your cursor at the start of the file (without selecting anything), and pressing the lightning bolt to run the entire file. ⚡ *All queries should run successfully one after another. If not, check to make sure you added semicolons ‘;’ after each query.*	*All 10 queries ran sequentially and were successful.*  # Time Action ✓ 9 13:15:53 select id, t... ✓ 10 13:15:53 select foru... ✓ 11 13:15:53 select foll... ✓ 12 13:15:53 select ad... ✓ 13 13:15:53 select out... ✓ 14 13:15:53 select pos... ✓ 15 13:15:53 select par... ✓ 16 13:15:53 select id fr... ✓ 17 13:15:53 select out... ✓ 18 13:15:53 SELECT ...

Late submission

Unless you have an approved extension (see below), **you will be penalised -10% of the maximum number of marks in the assignment per calendar day that your submission is late** ⚠. For instance, if you received a 78% raw score, but submitted 2 days late, you'd receive a 58% for the assignment.

Requesting a Submission Deadline Extension

If you need an extension due to a valid reason, you will need to provide evidence to support your request **the time indicated on the first page of these instructions**. Medical certificates need to be at least two days in length.

Requests received after the assignment deadline will be rejected, based on the requirements of the Special Consideration policy (https://eng.unimelb.edu.au/students/coursework/study-resources/extensions-and-special-consideration):

If students need to request an extension longer than 4 days or submit the request after the assessment submission deadline, the application must have appropriate supporting documentation and be submitted via the Special consideration portal.

(In other words, any extensions which fall within the scope/criteria of the Special Consideration policy needs to be assessed formally and centrally by University administration and not by the teaching team.)

To request an extension:

- Email the central subject inbox:

  INFO20003-semester2@unimelb.edu.au

  ... from your university email address, supplying your student ID, how many days you’d like to extend, and evidence that can support the number of days you are requesting. Please include in the subject [INFO20003 Assignment 2 Extension].

- If your submission deadline extension is granted you will receive an email reply granting the new submission date. **Do not lose this email!**

Reminder: INFO20003 Hurdle Requirements

To pass INFO20003, you must pass two hurdles:

- **Hurdle 1:** Obtain at least 50% (15/30) or higher for the three assignments (each worth 10%)
- **Hurdle 2:** Obtain at least 50% (35/70) or higher for the combination of quizzes and end of semester exam
  It is our recommendation to students that you attempt every assignment and every question in the exam.

Reminder: Academic Integrity

Students are kindly reminded to adhere to comply with the University's Academic Integrity Policies at all times.

Please read https://academicintegrity.unimelb.edu.au/plagiarism-and-collusion and please note:
"If a student is found to have deliberately plagiarised or colluded the penalties are severe and can include failure of a subject or exclusion from the University. The University provides extensive resources and educates students about academic integrity so that students are aware of what constitutes plagiarism and collusion, and the consequences of those practices."
(Source: https://academicintegrity.unimelb.edu.au/plagiarism-investigation-and-penalties)

GOOD LUCK!

Document Change Log for clarification:
V1.1 Uploaded 2 Sept 2024 – added disambiguations [Ed Post #345 #346 - i.e., Q5 Q6], typo fix and text clarification/position in diagram [Ed Post #350]
V1.2 Updated 4 Sept 2024 – corrected image 0-M at moderatorreport [Ed Post #361]
V1.3 Updated 5 Sept 2024 – very minor typo [Ed Post # 366 - may->many Q9]
V1.4 Updated 9 Sept 2024 – minor typo in text [Ed Post # 398 – Q9 refers to the disciplinaryAction flag]; Q8 phrasing disambiguated [via authoritative answer on Ed Post #385].