Marked Lab: Advanced SQL Queries

This lab must be prepared in teams of 1 or 2 students and submitted on Campus. You can check the due date in the submission area.

Contents

1 Work to Do

You have to answer the questions given in the script `queries.sql` in the form of SQL comments. The questions relate to the database created and populated by the script `database.sql`.

To do the lab: (1) download and execute the script `database.sql`, (2) download and complete the script `queries.sql`.

The population of the database given as an example in `database.sql` is deliberately simple. Some questions in the script might not yield any answer with this population. Therefore, you will need to modify the population in order to thoroughly test your queries.

The archive `Example.zip` gives you an example of the files you have to submit ("File to Submit" folder) in accordance with the resource files you are given ("Resources" folder).

If you have a question regarding the lab, please post it on the Questions & Answers forum. I will not reply to emails sent to my email address. Thank you.

2 Beware

2.1 Automatic Marking

Your script will be marked automatically. The answers it outputs will be compared line by line and column by column to those given by the solution script when executed on the same database. Therefore, you must be very careful about the following:

- Answer each question precisely, like in the labs: answer each question with only one SQL query; include all the requested attributes, and only those, in the specified order. If no specific attribute is requested, include them all, in the order they are listed in the table declaration.

- For some questions, the attributes involved in filtering or sorting must not be displayed: this is not a mistake. You can of course display these attributes when testing your queries, but do not forget do remove them before submitting your script.

- Be sure to use the right case for table and attribute names, as defined in the script `database.sql`. Beware: the database server I will use is case-sensitive, whereas the server of your local WAMP/XAMPP/MAMP installation might not be.

- Do not remove or alter any line of the script, especially the tags “select 'Query xy' …”: they tag the the question being answered in the output.

- Your script must output the result of the queries and nothing more (e.g. temporary results, comments, etc.); it must not call or include, in whole or part, the script `database.sql`.

The name of the columns of the tables output by your script is ignored during the marking process: in the select clause of your queries, you may rename attributes and expressions at will.

2.2 Executing your Script

I will run your script on my local MySQL installation using a tool similar to the `Import` tab of phpMyAdmin. Therefore, you must perform the same test to check that everything is correct before submitting your script.

Please note that testing your queries one by one using the SQL tab of phpMyAdmin is not enough. Running your script at once using the `Import` tab will allow you to detect the most common errors: (a) missing semicolon at the end of each SQL statement, (b) use of the wrong case for table and attribute names (recall that my test server is case-sensitive), (c) failure to execute `set session sql_mode…` when testing your queries one by one.

2.3 Marking Database

A query must not make any assumption about the population of the database: it must return a valid result regardless of the population. Your script will be tested against a database whose population is different from that of the example database. (Its schema, however, will of course be the same.)

Therefore, when asked about the products ordered by Smith (for example), your query must not use the hardcoded value 1234 just because 1234 is Smith’s ID in the example database. Rather, your query must use the information provided in the question (here “Smith”) and this information alone.

Be sure to address all special cases: null values, duplicate values, entities that do not participate in relationships, etc. If there is a doubt whether a tuple matches a question, especially because of null values, your query must not output that tuple, as the where clause does.

2.4 Standard SQL

MySQL accepts queries that are illegal with respect to standard SQL, especially with the `group by` clause. In order to detect such queries, the query script contains the following command:

```
set session sql_mode = 'ONLY_FULL_GROUP_BY';
```

Obviously, you must not remove this line from the script.

3 Marking Scheme

The tentative marking scheme is as follows:

Item	**Marks**
Queries	18
Code Quality	2
Total	20

All the queries, whether simple or complex, bear the same marks. As for the "code quality", your queries must (1) be written in standard SQL, (2) only use SQL92's "natural join" or "join on" for joins, (3) be properly formatted.

4 Submission

4.1 Deliverable

The deliverable consists of the script `queries.sql`, which you must rename as `LASTNAME1.FirstName1.LASTNAME2.FirstName2.sql`, **without any space character**, to identify the team members.

4.2 Submitting

You must submit the file into the submission area directly, **without creating an archive**. You may submit it again as often as you wish until the deadline, provided you **always do so under the same Campus user**.