School of Computing and Information Technology
University of Wollongong

Session: **Spring 2019**
Lecturer: **Janusz R. Getta**
Tianbing Xia

CSIT115/CSIT815 Data Management and Security
Laboratory 5

Published on 29 September, 2019

Scope

This laboratory includes the tasks related to the applications of advanced DML statements, and `SELECT` statements with `WITH` clause.

The outcomes of the laboratory work are due by **Saturday 12 October 2019, 7.00 pm (sharp).**

Please read very carefully information listed below.

This laboratory contributes to 3% of the total evaluation in a subject CSIT115 and it contributes to 3% of the total evaluation in a subject CSIT815.

A submission procedure is explained at the end of specification.

A submission procedure is explained at the end of specification.

This assignment consists of 1 task and specification of the task starts from a new page.

It is recommended to solve the problems before attending the laboratory classes in order to efficiently use supervised laboratory time.

A submission marked by Moodle as `Late` is treated as a late submission no matter how many seconds it is late.

A policy regarding late submissions is included in the subject outline.

A submission of compressed files (zipped, gzipped, rared, tared, 7-zipped, lhzed, … etc) is not allowed. The compressed files will not be evaluated.

All files left on Moodle in a state `"Draft(not submitted)"` will not be evaluated.

An implementation that does not compile due to one or more syntactical errors scores no marks and implementation that has the processing errors scores no marks.

It is expected that all tasks included within **Laboratory 5** will be solved **individually without any cooperation** with the other students. If you have any doubts, questions, etc. please consult your lecturer or tutor during lab classes or office hours. Plagiarism will result in a **FAIL** grade being recorded for the assessment task.

Prologue

In this laboratory you must use a virtual machine with MySQL 8.0.16. All technical details on how to start and how to use a virtual machine have been explained and practiced in Laboratory 1, task 2 and task 3.

Connect to Moodle and download the files `dbcreate.sql`, `dbdrop.sql`, `dbload.sql`, `dbcount.sql`, and `dbschema.bmp` from Sample database section on Moodle.

SQL script `dbcreate.sql` can be used to create the relational tables of a sample database. SQL script `dbdrop.sql` can be used to drop the tables of a sample database. SQL script `dbload.sql` can be used to load data into a sample database. SQL script `dbcount.sql` can be used to display the total number of rows in each table included in a sample database. Finally, a file `dbschema.bmp` contains a conceptual schema of a sample database.

Connect to MySQL database server either through command line interface mysql or graphical user interface SQL Developer.

When connected, select a database `csit115` with a command `use csit115`.

To create the relational tables of a sample database, process SQL script `dbcreate.sql`.

To load data into the relational tables created in the previous step process SQL script `dbload.sql`.

To list the names of relational tables created, use a command `show tables`.

To list a structure of a relational table `<table-name>` use a command `describe <table-name>`.

To list the total number of rows in each relational table process a script `dbcount.sql`.

Use an image viewer to open a file `dbschema.bmp` with a conceptual schema of the sample database.

No report is expected from the implementation of the actions listed above.

Tasks

Task 1 (3 marks)

Download a file `solution1.sql` and insert into the file the implementations of the following data definition, data manipulation, and query operations on the sample database. Note, that you have to use more than one SQL statement to implement the operations.

(1) Create a new relational table to store information about the driving licenses of all drivers together with total number of trips performed by each driver.

Enforce, the appropriate consistency constraints on the new table.

Next, copy into the new table information about the driving license numbers of all drivers, and the total number of trips performed by each driver.

Note, that all drivers who have not perform any trip yet must be included into the table with the total number of trips equal to zero (`0`).

When ready use `SELECT` statement to list the contents of the relational table created and filled with data in the ascending order of the total number of trips.

(2) Use **a single SQL statement** to create a relational table and to load into the table trip numbers and for each trip, a name of origin city. Next, enforce the appropriate consistency constraints on the new table.

For example, if the following rows are included in `TRIPLEG` table

```
1, 1, Sydney,     Melbourne
1, 2, Melbourne,  Sydney
1, 3, Sydney,     Perth
2, 1, Brisbane,   Sydney
3, 1, Brisbane,   Perth
3, 2, Perth,      Sydney
```

then the following rows must be inserted into the new table.

```
1, Sydney
2, Brisbane
3, Brisbane
```

(3) Add to a relational table `TRUCK` information about the total number of trips performed by each truck. Note, that if a truck has not been used for any trip so far then for such truck, the total number of trips must be set to zero (`0`). Enforce the appropriate consistency constraints on a relational table `TRUCK`.

Use `SELECT` statement to list the contents an extended relational table `TRUCK` in the ascending order of the total number of trips.

(4) Delete from the database information about all trips performed in `2014`. Information about all legs of the trips performed in `2014` and information about the origins of the trips performed in `2014` must be deleted as well. You are not allowed to drop and/or to suspend any referential integrity constraints. The deletions of the legs and origins of trips must be implemented as the nested `SELECT` statements.

(5) Implement the following query as `SELECT` statement with `WITH` clause.

Find the distinct names of cities visited by a driver during a trip number `1` and during a trip number `8`. Assume that a city is visited by a driver if it is an origin or a destination or an intermediate stop of a trip.

The query must be implemented as a sequence of subquery definitions following `WITH` keyword and ended with the final `SELECT`.

(i) The first subquery definition must find the distinct names of cities visited during a trip number `1`.

(ii) The second subquery definition must find the distinct names of cities visited during a trip number `8`.

(iii) The third subquery definition must find the distinct names of cities visited during both trips number `1` and number `8`.

(iv) The final `SELECT` statement must retrieve all rows created by the fourth subquery.

To create a report from processing of a script file `solution1.sql` open a Terminal window and start the command line interface `mysql` in the following way:

```
mysql -u csit115 -p -v -c
```

Next, process SQL script `solution1.sql` and save a report in a file `solution1.rpt`. Note, that when started with the options **`-v`** and **`-c`** the command line interface includes both listing of `SELECT` statements processed and the comments included in the original version of a file `solution1.sql`.

Deliverables

A file `solution1.rpt` with a report from processing of SQL script `solution1.sql`. The report must be created with the command line interface mysql, the report MUST NOT include any errors, and the report must list all SQL statements processed and all comments included in the original (downloaded) version of `solution1.sql`. Marks will be deducted for the missing comments. Submission of a file with a different name and/or different extension and/or different type scores no marks.

Submission
Note, that you have only one submission. So, make it absolutely sure that you submit correct files with the correct contents. No other submission is possible !

Submit a file **`solution1.rpt`** to Moodle in the following way:

(1) Access Moodle at **`http://moodle.uowplatform.edu.au/`**
(2) To login use a **`Login`** link located in the right upper corner the Web page or in the middle of the bottom of the Web page
(3) When logged select a site **`CSIT115/CSIT815 (S219) Data Management & Security`**
(4) Scroll down to a section **`Submissions`**
(5) Click at a link **`In this place you can submit the outcomes of Laboratory 5`**
(6) Click at a button **`Add Submission`**
(7) Move a file **`solution1.rpt`** into an area **`You can drag and drop files here to add them`**. You can also use a link **`Add`**...
(8) Click at a button **`Save changes`**
(9) Click at a button **`Submit assignment`**
(10) Click at the checkbox with a text attached: **`By checking this box, I confirm that this submission is my own work,`** ... in order to confirm the authorship of your submission
(11) Click at a button **`Continue`**

End of specification