School of Computing and Information Technology
University of Wollongong

Session: **Spring 2018**
Lecturer: **Tianbing Xia**

CSIT115/CSIT815 Data Management and Security
Laboratory 6

Scope

This laboratory includes a task related to SQL statements.

Important messages

Please read the messages listed below before implementation of a task included in a specification of Laboratory 6.

More implementation related information can be found in "How to … ?" Cookbook available through Moodle or at:
`http://www.uow.edu.au/~jrg/115/COOKBOOK`.

The outcomes of Laboratory 6 are due by **Saturday, 13 October, 2018, 11.55 pm (sharp).**

Laboratory 6 contributes to 3% of the total evaluation in the subject. 2 tasks are included in this laboratory.

A submission procedure is explained at the end of this document.

Only one submission of Laboratory 6 is allowed and only one submission per student is accepted. Please make sure that you submit the correct files.

A submission that contains an incorrect file attached is treated as a correct submission with all consequences coming from the evaluation of the file attached.

Compressed (zipped, rared, tared, etc) files will not be evaluated.

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

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 CSIT115/815 Subject Outline.

Prologue

Download the files `dbcreate.sql`, `dbdrop.sql`, `dbload.sql`, template files `solution1.sql` and `solution2.sql`, and `dbschema.pdf`. Copy the files to your USB drive or email the files to yourself such that you can access all of them either through command line interface mysql or graphical user interface MySQL Workbench.

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

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

```
use csit115
```

It is strongly recommended to process SQL script file `dbdrop.sql` to drop all relational tables used for implementation of the previous assessment task.

To re-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>`.

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

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

Tasks

Task 1 (1 mark)

Process SQL script file dbdrop.sql to drop all relational tables used for implementation of the previous task.

To re-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.

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

Implement the queries by create at least two views for each of the questions in a given template script file solution1.sql for the questions below.

(1) Find employee number and name for the employees who work on the same total number of projects as the employee '00101' does.

(2) Find employee number and name for the employees who work in the department that managed by Alvin.

When ready execute SQL scrip solution1.sql and save a report from execution in a file solution1.rpt.

Start mysql command line in a terminal by

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

Then input a password csit115. It allows MySQL database server to display comments and SQL script for each question in a report.

Use the commands as follows to generate a report for Task 1.

```
tee solution1.rpt;
source solution1.sql;
notee;
```

Hint: You can find similar statements already implemented in the "COOKBOOK".

Deliverables

A file solution1.rpt with a report from processing of SQL script solution1.sql. The report MUST have no errors and the report MUST list all SQL statements processed.

A report that contains no listing of executed SQL statements scores no marks and report that contains errors also scores no marks !

Submission of a file with a different name and/or different extension and/or different type scores no marks.

Task 2 (2 marks)
Implement the following advanced manipulations on the database in SQL in a given template script file `solution2.sql` for the questions below.

(1) Create a database with the same name as a *prefix of your University email account*. For example, if your University email account is `xyz007@uow.edu.au` then a name of a database must be `xyz007`.

(2) Create two users with the user names: *tec,* and *admin.* The passwords are up to you.

(3) While connected as a user `root`, process the scripts `dbcreate.sql` and `dbload.sql` to create and to load data into the relational tables later on used in this laboratory class. All relational tables must be located in a database created in step (1). SQL statements processed by the scripts `MUST NOT` be included in a report from processing of a script `solution1.sql`. It means that before processing of the scripts you must execute `notee` statement to turn off spooling and after processing of the scripts `tee solution1.rpt` to turn on spooling into a report file.

(4) Next, grant the read and write privileges on entire database *prefix of your University email account* to a user *admin*. The privilege must be granted such that a user *admin* is not allowed to grant the same privilege to another user.

(5) Next, grant a read privilege on a relational table `Department` and `Project` located in a database *prefix of your University email account* to a user *tec*. The privilege must be granted such that a user *tec* is able to grant the same privileges to the other users.

(6) Next, grant the write privileges on a relational table `WorksOn` located in a database *prefix of your University email account* to a user *tec*. The privileges must be granted such that a user *tec* is not able to grant the same privileges to the other users.

(7) Next, grant a privilege to create relational tables located in a database *prefix of your University email account* to a user *admin*. The privilege must be granted such that a user *admin* is allowed to grant the same privilege to another user.

(8) Finally, lists all privileges granted to the users *tec*, and *admin*. The script must use data dictionary views included in `mysql` database to list the privileges.

When ready execute SQL scrip `solution2.sql` and save a report from the processing of the script in a file `solution2.rpt`.

Start mysql command line in a terminal by

```
mysql -u root -p  -v  -c
```

Then input a password `csit115. It` allows MySQL database server to display comments and SQL script for each question in a report.

Use the commands as follows to generate a report for Task 2.

```
tee solution2.rpt;
source solution2.sql;
notee;
```

Hint: You can find a lot of applications of database definitions and manipulation statements in the "COOKBOOK".

Deliverables
A file `solution2.rpt` with a report from processing of SQL script `solution2.sql`. The report MUST have no errors and the report MUST list all SQL statements processed.

A report that contains no listing of executed SQL statements scores no marks and report that contains errors also scores no marks!

Submission of a file with a different name and/or different extension and/or different type scores no marks.

Submission

Submit files **`solution1.rpt`** and **`solution2.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/DPIT115/CSIT815 (S218) 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 6`**

(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) Repeat step (7) for the file **`solution2.rpt`**.

(9) Click at a button **`Save changes`**

(10) Click at a button **`Submit assignment`**

(11) 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

(12) Click at a button **`Continue.`**

It is expected that a problem included within **Laboratory 6** 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 that assessment task.

End of specification