

CSIT115/CSIT815 Data Management and Security
Laboratory 6

Published on 10 October, 2019

Scope

This laboratory includes the tasks related to discretionary access control in relational database systems.

The outcomes of the laboratory work are due by **Saturday 19 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.

This laboratory work consists of 1 task.

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 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 the assessment task.

Prologue

Connect to Moodle and download the files `dbcreate.sql`, `dbdrop.sql`, `dbload.sql`, `dbcount.sql`, and `dbschema.pdf` 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.pdf` contains a conceptual schema of a sample database.

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`.

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 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 actions listed above.

Tasks

Task 1 (3 marks)

Your task is to implement and to process SQL script `solution1.sql` that creates a new database, creates the new user accounts, creates the new roles, grants roles and privileges to the new user accounts, sets resource limits and locks the accounts. Insert into a file `solution1.sql` implementation of the steps listed below. Note, that a user `csit115` does not have the privileges required to process these steps. You must connect as a user `root` with a password `csit115`. The steps to be implemented are the following.

- (1) Create a database with the same name as a prefix of your University email account. For example, if your University email account is `jrg@uow.edu.au` then a name of a database should be `jrg`. 0.1 mark
- (2) Create two new user accounts. The names of user accounts and the passwords are up to you. 0.1 mark
- (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). A listing of 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 script you must process `notee` statement to turn the spooling off and after processing of the scripts you must process a statement `tee solution1.rpt` to turn the spooling on into a report file. 0.2 mark
- (4) Next, create two new roles: `driver` and `admin` and grant to a role `admin` the read privileges on the entire database. The privileges must be granted such that any owner of a role `admin` cannot grant the same privileges to another role or user. 0.2 mark
- (5) Next, grant to a role `driver` a read privilege on a relational table `DRIVER` located in the database. A privilege must be granted such that any owner of a role `driver` can grant the same privilege to another role or user. 0.2 mark
- (6) Next, grant to a role `driver` the read and write privileges on the relational table `TRIPLEG` located in the database. The privileges must be granted such that any owner of a role `driver` cannot grant the same privilege to another role or user. 0.2 mark
- (7) Next, grant to a role `driver` a read privilege on the columns `FNAME`, `INITIALS`, `LNAME` in a relational table `EMPLOYEE`. A privilege must be granted such that any owner of a role `driver` cannot grant the same privilege to another role or user. 0.3 mark

- (8) Next, grant to a role `admin` an insert privilege on a relational table `EMPLOYEE`. A privilege must be granted such that any owner of a role `admin` can propagate the same privileges to another role or user 0.2 mark
- (9) Next, grant to a role `admin` a privilege to create relational tables located in the database. The privileges must be granted such that any owner of a role `admin` cannot grant the same privileges to another role or user. 0.2 mark
- (10) Next, grant to a role `admin` a privilege to create relational views located in the database. The privileges must be granted such that any owner of a role `admin` cannot grant the same privileges to another role or user. 0.2 mark
- (12) Next, grant to a role `admin` a read privilege on information about the trips completed in 2018. A hint is on create a relational view and grant a read privilege on the view. A privilege must be granted such that any owner of a role `admin` cannot grant the same privilege to another role or user. 0.5 mark
- (13) Next, grant a role `driver` to a role `admin` and then grant a role `admin` to one of the users created in step (2) and a role `driver` to another user. 0.2 mark
- (14) Next, set a resource limit on maximum total number of concurrent connections available to both users created in step (2). The maximum number of concurrent connections is up to you. 0.3 mark
- (15) Finally, lock the accounts of the new users. 0.1 mark

To implement and to test SQL script file `solution1.sql` you can either use graphical user interface MySQL Workbench or command line interface `mysql`.

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 root -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 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) 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