School of Computing and Information Technology
University of Wollongong

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

CSIT115/CSIT815 Data Management and Security
Assignment 2

Published on 9 September 2019

Scope

This assignment is related to implementation of simple `SELECT` statements, `SELECT` statement with `GROUP BY` and `HAVING` clauses, `SELECT` statements that join and antijoin relational tables, nested `SELECT` statements with set membership operation, and nested `SELECT` statements with `EXISTS/NOT EXISTS` clauses.

Please read very carefully information listed below.

This assignment contributes to 8% of the total evaluation in a subject CSIT115 and it contributes to 8% of the total evaluation in a subject CSIT815.

The outcomes of the assignment work are due by **Saturday 28 September 2019, 7.00 pm (sharp).**

A submission procedure is explained at the end of specification.

This assignment consists of 3 tasks and specification of each 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 **Assignment 2** 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 assignment 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.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.bmp` 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 (2 marks)

Download a file `solution1.sql` and insert into the file the implementations of the following queries as `SELECT` statements of SQL.

(1) Find the first and last names of employees who live in NSW, Hurstville, Victoria St. or NSW, Pennant Hills.

(2) Find the registration numbers, capacities, and statuses of all truck whose capacity is in range between `2000` and `4000` both inclusive.

(3) Find the first and last names of employees who did not provide information about initials and who provided information about a date of birth.

(4) Find the first and last names of all employees born either in `1980`, or `1992` or `1992` It is your task to find an appropriate standard function that can be used to extract a year from a date.

(5) Find the registration numbers of all trucks. The results should be displayed in the ascending order of the statuses and for all trucks that have the same status the results should be displayed in the descending order of registration numbers.

(6) Find the total number of distinct names of cities being the departure locations for at least one trip. A city is a departure location when it is a departure location of the first leg of a trip.

(7) Find the total number of trips performed by each driver. List the driving licence numbers associated with the total number of trips. Ignore the drivers who performed no trips so far.

(8) Find the total number of times each truck has been used for the trips. List the truck registration numbers associated with the total number of trips. List only the trucks used more than 5 times.

(9) Find the trip numbers, driving licence numbers of the drivers who performed the trips and the registration numbers of the trucks used for the trips done in the first 30 days after 1 February 2016.

(10) Find the total number of trips that started or finished or passed through a city that has a letter `y` in its name.

When ready process a script file `solution1.sql` with `SELECT` statements.

To create a report from processing of `SELECT` statements 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.

Task 2 (3 marks)
Download a file `solution2.sql` and insert into the file the implementations of the following queries as `SELECT` statements of SQL.

The queries listed below must be implemented as `SELECT` statements with `JOIN` or `OUTER JOIN` operation.

(1) Find the first and the last name of the drivers who are on leave at the moment.

(2) Find the registration numbers of trucks that are used just now and that have been used for at least one trip in 2016.

(3) Find the first and the last names of drivers who performed at least one trip in `January 2016`. It is your task to find an appropriate standard function that can be used to extract a month and year from a date.

(4) Find the first and last names of employees who are not drivers.

(5) Find the first and last name of drivers who performed no trips so far.

(6) Find the total number of trips performed by each driver. List the driving licence numbers associated with the total number of trips. Do not ignore the drivers who performed no trips so far.

When ready process a script file `solution2.sql` with `SELECT` statements.

To create a report from processing of `SELECT` statements 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 `solution2.sql` and save a report in a file `solution2.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 `solution2.sql`.

Deliverables
A file `solution2.rpt` with a report from processing of SQL script `solution2.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 `solution2.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.

Task 3 (3 marks)
Download a file `solution3.sql` and insert into the file the implementations of the following queries as `SELECT` statements of SQL.

The queries listed below must be implemented as nested `SELECT` statements with `IN/NOT IN` set membership operation.

(1) Find the first and the last name of the drivers who are on leave at the moment.

(2) Find the registration numbers of trucks that are used just now and that have been used for at least one trip in 2016.

The queries listed below must be implemented as nested queries with `EXISTS/NOT EXISTS` clauses.

(3) Find the first and last names of employees who are not drivers.

(4) Find the first and last name of drivers who performed at least one trip in 2017.

A query listed below must be implemented with a set algebra operation.

(5) Find the distinct names of cities visited by a driver during a trip number `1` or 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.

A query listed below must be implemented as a nested query.

(6) Find the distinct names of cities visited by a driver during a trip number `1` and a trip number `8`. Note, that we try to find the cities visited during both trips number `1` and 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.

When ready process a script file `solution3.sql` with `SELECT` statements.

To create a report from processing of `SELECT` statements 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 `solution3.sql` and save a report in a file `solution3.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 `solution3.sql`.

Deliverables

A file `solution3.rpt` with a report from processing of SQL script `solution3.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 `solution3.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 the correct files with the correct contents and correct types. No other submission is possible !

Submit the files **`solution1.rpt`**, **`solution2.rpt`**, and **`solution3.rpt`** through 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 and Security`**
(4) Scroll down to a section **`Submissions`**
(5) Click at a link **`In this place you can submit the outcomes of Assignment 2`**
(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 files **`solution2.rpt`** and **`solution3.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`**

End of specification