COMPSCI/SOFTENG351 and COMPSCI751
Lab 1

Lab material is provided for students to become familiar with using SQL and working with the most popular open-source relational database management system MySQL.

SQL Syntax and Conventions.

Best Practice:

Except where testing for equality, MySQL ignores upper and lowercase. Many texts use the documentation style of putting all SQL in UPPERCASE, and variables in lower case, alternatively SQL commands in lowercase, operators and table names in UPPERCASE. It is good practice to develop a similar standard style for readability/usability although MySQL does not care, it stores table and column names in its data dictionary in lowercase and converts what you type to lowercase before checking for them. The underscore _ is used by convention to link words in names, eg USER_TABLES. MySQL allows table and column names up to 64 characters long. This allows you room to create full, unambiguous and meaningful names. Avoid abbreviations and plurals.

Basic SQL Queries.

The basic form of an SQL query is as follows:

```
SELECT [DISTINCT] {*| column_name (, column_name,…)}
FROM table_name [alias] (, table_name,…)
[WHERE condition]
[GROUP BY column_list] [HAVING condition]
[ORDER BY column_list];
```

Today we will only be concerned with the most basic form:

```
SELECT {*| column_name (, column_name,…)}
FROM table_name;
```

SELECT specifies which columns are to appear in the output
FROM specifies the tables to be used
* is shorthand for ALL. Used when you want to view all the columns in a table.

Getting Started.

By now you should have received an email from the database administration help@sit.auckland.ac.nz with your login details to the Alternate MySQL server (MySQL i5.5.27). Go to **https://www.sit.auckland.ac.nz/student_database/** and log in. It is a good idea to change your password. For this purpose, choose from the **Actions** menu **Change password**, type in and re-type a new password of your choice. *In case you find yourself at the login menu, simply press the refresh button on your web browser and you are back at the main menu.* We have created a MOVIES database that will serve as a running example in the labs throughout the first part of the course. From the phpMyAdmin menu on the left, select the database. The name of the database is different from student to student. It has one of the three formats

stu_UPI_COMPSCI_351_C_S1_2017
stu_UPI_SOFTENG_351_C_S1_2017
stu_UPI_COMPSCI_751_C_S1_2017

where UPI shows your own upi. In what follows, we will simply refer to the database as MOVIES, and you will need to replace MOVIES by one of the three strings above that applied to you. The database schema for MOVIES is available from CANVAS. The following is a list of the tables in MOVIES:

MOVIE
PERSON
AWARD
RESTRICTION_CATEGORY
DIRECTOR
WRITER
CREW
ROLE
SCENE
RESTRICTION
APPEARANCE
MOVIE_AWARD
CREW_AWARD
DIRECTOR_AWARD
WRITER_AWARD
ACTOR_AWARD

To define the MOVIES database schema and to populate it, you will need to execute the following SQL scripts available on CANVAS: create_all.sql, movie_table.sql, person_table.sql , award_and_restriction_cat.sql, staff_tables.sql, movie_awards.sql, scene_appearance.sql.

Note that the SQL scripts cannot be executed in any arbitrary order. Why not?

To load an SQL script you can either **Import** the scripts above, or copy-and-paste their content into the **Run SQL query/queries window** after selecting the **SQL** button. To execute a script just press **Go** at the bottom right.

Now, the MOVIES database should be up and running and you can start accessing the data in it by using the SQL query language.

Data Dictionary

The data dictionary stores all of the information that is used to manage the objects in the database. It is a source of valuable information for the database administrator, developers and end users. The Database named **INFORMATION_SCHEMA** contains descriptions for the objects that make up the data dictionary. Try the following:

```
SELECT * FROM INFORMATION_SCHEMA.TABLES;
```

To get a listing of the tables created in the MOVIES database use:

```
SELECT TABLE_NAME
FROM INFORMATION_SCHEMA.TABLES
WHERE TABLE_SCHEMA = 'MOVIES';
```

To see how the columns were defined in the tables, tell MySQL to describe the table:

```
Describe MOVIE; (or Desc MOVIE)
```

This displays a descriptive table that lists the columns and their definitions for the MOVIE table.

Error Handling

If you run an incorrect statement MySQL will stop on encountering the first error. It will pinpoint the line at which the error occurs and the offending word. Try and solve the errors before asking for help.

Simple Queries

Try some simple queries on the MOVIES database within the **SQL > Run SQL query/queries window** and get a feel for what MySQL is doing with your commands.

1. List the contents of the MOVIE table.
2. List all American movies.
3. List all directors over 40 years of age.
4. List all directors of New Zealand movies who are over 40 years of age.
5. List all movies for which the director and the writer were the same person.

When you have finished a session select the EXIT button from the phpMyAdmin menu to close MySQL.