Practical 1

Working with MySQL in Linux and creating tables

Learning objectives

1. Initiating working with MySQL database in Linux environment using commands
2. Be familiar with the workflow to create SQL commands in a text file and use them in the Terminal
3. Create simple tables in an existing database
4. Getting information about a database
5. Be familiar with the online MySQL documentation

Tasks

In this practical, you will learn how to work with the MySQL interactive environment. You will be using Linux environment and use the Terminal to enter commands to work with MySQL.

1. Set up your working directory

In Curtin labs, log in to the computer using your Oasis username and password. Click on the 'VMWare Horizon client' icon on your desktop and provide your Oasis username and password again to connect to the virtual machine.

You will see 'mydesktop.curtin.edu.au' icon and click on that.

Then select 'Computer Science Linux Lab' icon and you will connected to the Linux environment.

Now, make sure you go to i**: drive** before starting your work.

In **i: drive**, get a Terminal (press CTRL+ALT+T)

Create a directory to do your DBS work (`> mkdir DBS`)

Go inside DBS directory (`> cd DBS`) and create a directory for each week

(`> mkdir Prac01 Prac02`)

In your `Home / i: drive you should have following directory structure`

DBS

- Prac01
- Prac02
- ..

Now, go to Prac01 directory to start the work (`>cd Prac01`)

If you are using your own PC and accessing a Linux environment, still make sure you have created a DBS directory and sub-directories to do your practicals. If you organize your work now, it would be much easier in later practicals.

2. Use terminal and starting MySQL server

You will use Terminal and run commands to access MySQL server and working with databases. It is a good practice to type your commands in a text editor and then copy and paste the command in the Terminal prompt. If you have saved your commands in a text file, you can re-run them and get the required output again easily.

Open a file in Vim (`> vim Prac01`) or any text editor you wish to use (Make sure you are in I: drive, and the `DBS/Prac01` directory)

You can keep the Terminal and the Text editor side by side so that work can be done easily.

In your Terminal prompt type the following and press enter key:

```
> mysql –u me -p
```

You will be prompted to give the password for the server.

Password is: ‘ myUserPassword’. Type this password and press Enter.

You will be connected to the MySQL server and the prompt will be changed to:

```
mysql>
```

You are now in the MySQL interactive environment.

To close the connection, type exit:

```
mysql> exit
```

Again connect to the MySQL server using the required command.

3. See the available databases and creating tables

It is a good practice to type your commands in a text editor and then copy and paste the command in the Terminal prompt. If you have saved your commands in a text file, you can re-run them and get the required output again easily.

Open a file in Vim (`> vim Prac01`) or any text editor you wish to use (Make sure you are in I: drive, and the `DBS/Prac01` directory)

You can keep the Terminal and the Text editor side by side so that work can be done easily.

1. To see the databases already in the server use the following command.

```
mysql> SHOW DATABASES;
```

2. Type the command in simple letters and observe the output.

```
mysql> show databases;
```

3. You will create tables in the `'dswork'` database in the following tasks.

   Use the `'dswork'` database by typing:

```
mysql> USE dswork;
```

4. Students, Units and Enrolments are three relations to be created in the 'dswork' database.

   You will see a message,' Database changed', indicating the working database is 'dswork'

Consider the following definition for Student relation.

Student relation

COL NAME	TYPE	SIZE	DESCRIPTION
SNo	CHAR	8	Student number
FirstName	VARCHAR	12	First name
LastName	VARCHAR	15	Last name
PhoneNo	CHAR	10	Student's telephone number

Type the following command in your text file. Then copy and paste the command to mysql prompt to create the Students table.

```
Mysql> CREATE TABLE Students (
sno  CHAR(8),
firstname VARCHAR(12) ,
lastname  VARCHAR(15) ,
phoneno  CHAR(10)
);
```

5. Write down the command to create the following table in the text editor(Hint : use CREATE TABLE statement)
   Use the command you have written to create `Units` table `in 'dswork'` database.

Units table			
COL NAME	**TYPE**	**SIZE**	**DESCRIPTION**
UnitIndex	CHAR	8	Unit code
Dept	CHAR	10	Department Code
UnitName	VARCHAR	40	Unit Name

6. Similarly, create the `Enrolments in 'dswork'` database using the following definition for the Enrolement table.

Enrolments table			
COL NAME	**TYPE**	**SIZE**	**DESCRIPTION**
Unit	CHAR	8	Unit code
Student	CHAR	8	Student number
Year	CHAR	4	Year enrolled
Mark	INT	3	mark for unit
Grade	CHAR	1	grade for unit

4. Getting information about your database

1. Display all the tables currently in your database using the following command.

```
SHOW TABLES;
```

2. To learn about the attributes of relation `Students`, issue the command:

Hint: The DESCRIBE command can be shortened to DESC.

```
DESCRIBE Students;
```

3. You can also use SHOW to see the contents of a table.

```
SHOW COLUMNS FROM Students;
```

5. Save the commands and reuse

1. Create a copy of the '`Prac01`' file as '`Pra01tables`' (Use `cp` command)
2. Open the '`Prac01tables`' file and delete all the commands except three commands to create three tables.
3. Change the Table names on the command to 'TestStudents' , ' TestUnits' and 'TestEnrolments'. Save the `Prac01tables` file.
4. Instead of copy and paste the command , you can use the file name to run the commands in the file . Type the following command and observe the output.

```
mysql> SOURCE Prac01tables;
```

5. Use a suitable command and see the tables in the databse, You will see three new tables are created.

6. On-line Documentation

The on-line documentation for the MySQL database system can be viewed at the URL: http://dev.mysql.com/doc/. This would be the first place to seek additional information about a command or ways to do a particular task.

Explore the documentation and find information about some of the commands you have used today.

Make sure you have saved all the commands you have created today in the DBS/Prac01/Prac01 file. We will be using the same commands in next week as well.

Check whether you have achieved learning objectives:

I am confident that I can,

Connect to MySQL server in Linux environment using Terminal and close the connection after my work	✓
Create a simple table using SQL commands	
Get information about a database using SQL commands	
Save SQL commands in a text file and reuse them by copy and paste/ reading from the file	
Use online MySQL documentation to know more about a topic/command	

Please refer lecture slides, reading materials, and online resources and attempt again, if all the objectives were not achieved. Ask your tutor and get help if you need any clarification.

It's always a good practise to try to finish the practical of a particular week, before attempting the next practical worksheet as your work will be building upon the previous week's tasks.

NOTE 1: Remember to use a text file and save commands

- Remember to first type the required commands in the text editor and then copy and paste it in the mysql > prompt.
- Save the text editor with the commands using suitable file name (e.g.: Prac01) so that you will have all the commands of the practical saved in a text file.
- You may use # and add comments, brief description of what you are doing before the command.
- Initially you will use short commands but in few weeks you will start creating lengthy commands so you will see the value of using of using a text editor. Additionally, all the commands you created in the practical as answers to the tasks can be saved using the text editor. Later you will learn to run them directly from the file also.

NOTE 2: Missing semi-colon error

- Every command in MySQL must be terminated with ";". If you forget this and press 'Enter', you'll receive a blank line instead of the usual prompt. When that happens, just type a semi-colon and Enter.

NOTE 3: naming conventions

- Note that the table name (Students) is capitalized while the attribute names (like sno) are not. This is a convention used in this unit to make it easy to tell them apart.
- Table names are case sensitive on the server being used, so take care to get this correct. Case doesn't matter for the attributes, but keeping them all in lower case helps differentiate them from table names and reserved words.
- We write reserved words in ALL-CAPS, again just to make them more obvious. MySQL will accept "Create Table", "CREATE TABLE" or "create table" equally (without the quotes).
- Another convention to make the table names plurals (*i.e.*, Students rather than Student) because a table contains many students, not just one. I also tend to do this with tables that contain relationships rather than entities, although this is less clear-cut.