Advanced Data Management (CMM524)
Solution/Discussion to Laboratory #2: Introduction to MySQL

1. Aims

- To familiarise with the virtual machine environment used in the labs.
- To use simple SQL commands to manipulate a database.

2. Outcomes

In completing this exercise, you should be able to:
- Use the CMM524 virtual machine environment in VMWare.
- Connect to the MySQL server running in the virtual machine.
- Do simple data manipulation and retrieval using SQL.

3. Using the CMM524 Virtual Machine

In our labs, we are using a virtual machine with MySQL installed.

- Run *VMWare WorkStation Player* in Windows.
- Click "*Open Virtual Machine*". Navigate to `D:\VMs\CMM524(v1.1)` until you see `CMM524VM (for students).vmx`
- Open the file and start the VM.
- Log into the VM using the following credentials:
  - **username: `training`**
  - **password: `training`**

Notes:
- The VM is only available on PCs in lab N527, N528, and N529.
- MySQL runs locally in the virtual machine which in turn is local to the PC you use. That means the data you store into MySQL stays in the PC and it not accessible from another machine.
- As all VMs are protected by the same password which everyone knows. **Do not store any confidential data in these VMs** as everyone in the class can access it. Delete and purge all data/files that you don't want others to see.
- Also, please do not vandalise these VMs.

4. Connecting to MySQL

The CMM524 VM has MySQL installed. To connect to it you need a client.

- Run a terminal.
  - If you don't know how to open a terminal in Ubuntu, see: https://itsfoss.com/open-terminal-ubuntu/
- In the terminal, run the simple mysql client:

```
mysql --user=training --password
```

- MySQL will prompt you to enter the password[1]. The password for the `training` account is also "`training`".
- Once you are logged in, what command do you use to show all databases available to you?

    - `SHOW DATABASES;`

4.1. Creating a Database

You can now create a database to store data of our online shopping domain.

- Create a database `OnlineShop`.
    - What is the SQL statement you use to create the database?
        - `CREATE DATABASE OnlineShop;`
    - What is the SQL command to show all databases in the system?
        - Already asked in the last section! You should now see the new database you just created.

4.2. Selecting a Database

To select/use the `OnlineShop` database, use the command:

```
USE OnlineShop;
```

4.3. Creating the Product Table

Use the following SQL command to create the `product` table in the current database:

```
CREATE TABLE product (
        code int,
        description varchar(255) NOT NULL,
        price float,
        PRIMARY KEY (code)
);
```

- Show all tables in your current database.
    - How many tables do you have?
        - You should see 1 table in the current database.
- What is the SQL command to show the structure of the `product` table?
    - `DESCRIBE product;`

4.4. Adding Data

With a `product` table created, you can start storing data. Here are some sample products:

1 If you are lazy, you can enter the password in the command line using "`mysql --user=… --password=...`". However, your password may show in a multi-user system if another user lists all processes running.

Code	Description	Price
1	Andrex rolls 54 rolls	22.80
2	Cusheen luxury 60 rolls	16.98
3	Andrex classic 16 rolls	6.50
4	Velvet comfort 45 rolls	27.99

- Populate the `product` table with the data above.
  - What SQL statement do you use to insert data into the table?
    - Use the INSERT command for each product. For example (in 1 line!!!):

```
INSERT INTO product (code,description,price)
VALUES (1,'Andrex rolls 54 rolls',22.8);
```

4.5. Retrieving Data

With some sample product data added:

- What is the SQL statement to retrieve all product data?
  - Use the SELECT command:
  - `SELECT * FROM product;`

- What is the SQL statement if you only want to get the product descriptions and prices only?
  - Use the SELECT command but specify the columns you want:
  - `SELECT description, sellingPrice FROM product;`

4.6. Modifying a Table

Assuming that you want to keep track of the stock level of each product, which default to 0 if not entered.

- Modify your `product` table to store the stock level of each product.
  - What SQL statement do you use to modify your table structure?
    - Use the ALTER command to add an extra column:
    - `ALTER TABLE product ADD COLUMN stockLevel int DEFAULT 0;`
- Show the structure of your table after the modification.
  - Use the DESCRIBE command. You should see the product table with the extra column now.

4.7. Dropping/Deleting a Table

Once you have completed the exercises above, you can delete the product table.

- List all tables in your current database.
- Drop/delete the `product` table.
  - What is the SQL command to drop the `product` table?
    - `DROP TABLE product;`

- List all tables in the database after deletion. Do you notice any difference?
  - The `product` table should be gone now.

4.8. Dropping/Deleting a Database

An easy way to delete all tables in a database is to delete the database itself. However, do this with caution as it will delete all tables in the database.

- Your `OnlineShop` database should now be empty.
- Show all databases available to you.
- Drop the `OnlineShop` database.
  - What is the SQL command to use?
    - `DROP DATABASE OnlineShop;`
- Show all database after dropping the OnlineShop database. What do you notice?
  - The `OnlineShop` database is gone.