Facilitator Meeting

Instructions 9 – Introduction to MySQL

Aims:

- Practice the methods to access the MySQL database.
- Practice the SQL commands by creating and manipulating a database table on the MySQL database server **feenix-mariadb.swin.edu.au**

The tasks are due next week by your facilitator meeting. Email the link of your web page running on the Mercury server to your facilitator before the due date to be marked off. Tasks will not be marked if the email is not received.

Getting Started:

Create a new folder '**lab09**' under the unit folder on the mercury server ~/*{your unit code}/www/htdocs* folder. Save today's work in this lab09 folder.

All Web pages should conform to HTML5 and must be validated.
You could also create and link an external stylesheet, to the pages, and this should be valid CSS3.

Task 1: How to login to your MySQL account:

(See also https://feenix.swin.edu.au/help/index.php?page=MySQL%20%28MariaDB%29)

1. Login to the **mercury.swin.edu.au** server using the 'PuTTY' client, with your SIMS user name and password.

2. To access your mysql account type in:
   mysql *(press Enter key)*

   The configuration file on **mercury.swin.edu.au** ensures that you are prompted for a password and to connect to your mysql account on the **feenix-mariadb.swin.edu.au** server.

   For most users, your **mysql username** will be the same as your **mercury username** and your **mysql password** will have been set initially to your date of birth in **ddmmyy** format.
   Press 'Enter' key after typing in your password.

   Your mysql password is not connected to your SIMS password.
   It is recommended that you change your mysql password and ***do not use*** the same password for your mysql account, as for other accounts, as later you will need to use your mysql password within PHP scripts. (See "How to change your mysql password" on the next page.).

3. You will arrive at the 'MySQL Monitor' or mysql command line client:
   MariaDB> …

4. On our mysql server, a database has already been created for you to use in labs and assignments, named as username_db where username is **s<7 or 10 digit Swinburne id>**, for example, s1234567 and s1234567_db.
   Access your database as follows:
   MariaDB> USE username_db;
   You will receive a confirmation message "Database changed".

5. The MariaDB SQL manual can be found here:
   https://mariadb.com/kb/en/mariadb/sql-structure-and-commands/ *... or see other simpler guides* ☺

How to change your MySQL password:

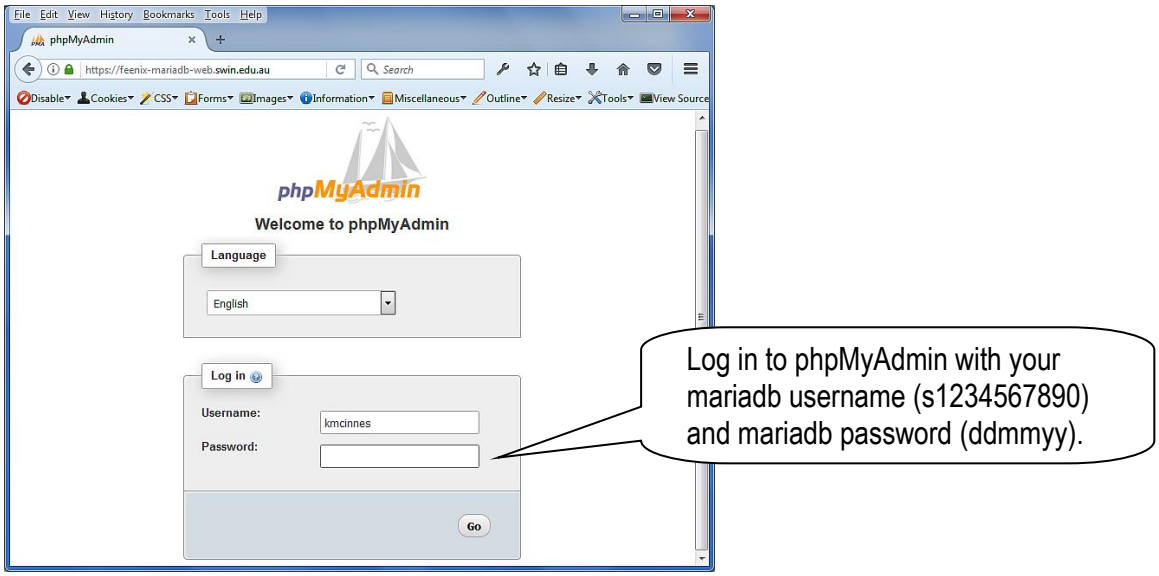
```
MariaDB> SET password=PASSWORD('newpwd');
```

Where `newpwd` represents the new password you would like to use.

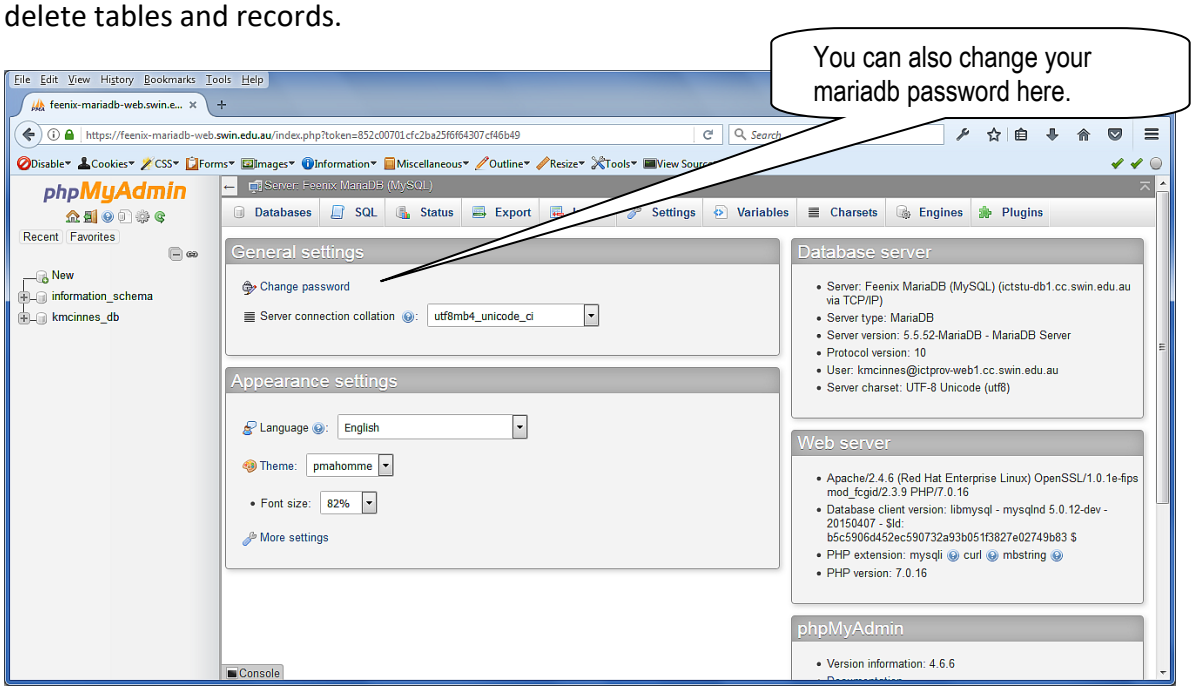
Remember that in Labs and in Assignments, you will have to **include (encode)** your database **password** in **all** PHP files that will access the database.
So **do not use** your SIMS or your mercury password, as your mysql password.

Task 2: Using phpMyAdmin

Instead of using the 'MySQL Monitor' command line interface, you may find it easier to use the MySQL web interface called **phpMyAdmin** which is available at https://feenix-mariadb-web.swin.edu.au/



From here you can manage the settings for phpMyAdmin and your database, and create, modify and delete tables and records.



Task 3: Creating a table

Note that if you managed your own website, you would need to first create a database before you would be able to create tables in your database. On our mysql server, a database has already been created for you.

Using your existing database '**s<Swinburne id>_db**', create a new table **cars** for a used car dealership.

CREATE TABLE cars .....;

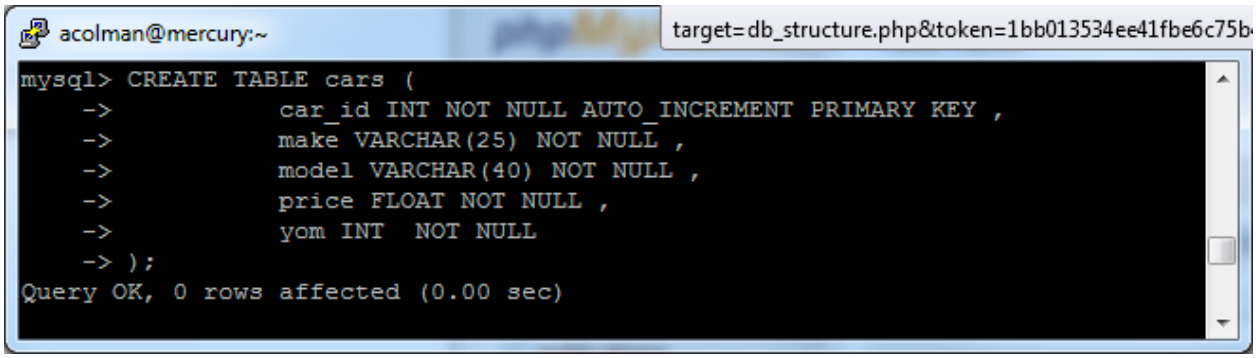
Include the following fields in the **cars** table with the following data types:

car_id	AUTO_INCREMENT PRIMARY KEY
make	string up to 25 characters – required
model	string up to 25 characters – required
price	float – required
yom (year of manufacture).	integer - required

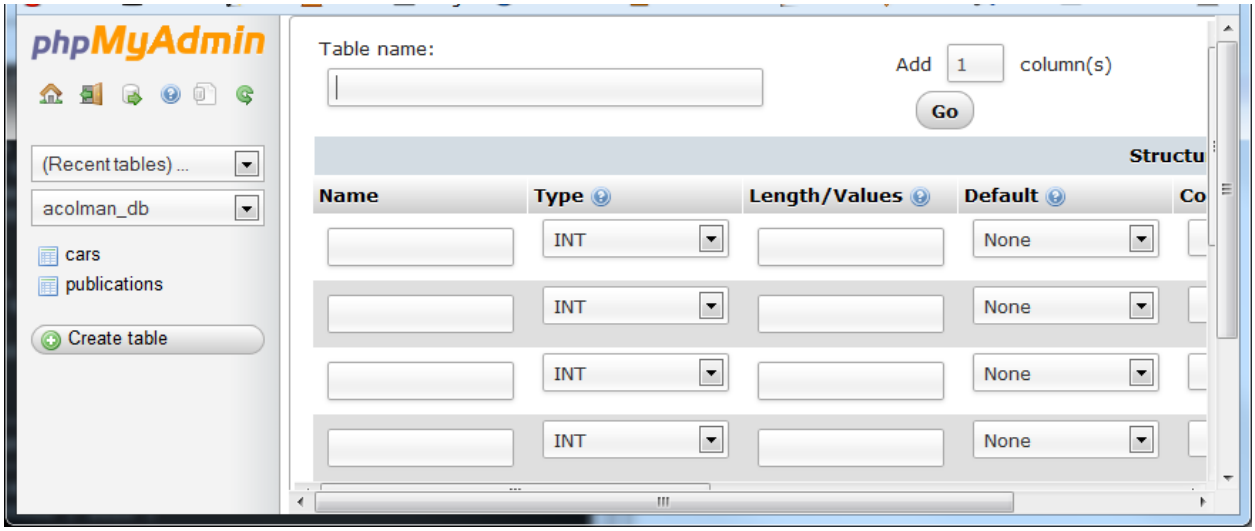
```
CREATE TABLE cars (
          car_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY ,
          make VARCHAR(25) NOT NULL ,
          model VARCHAR(40) NOT NULL ,
          price FLOAT NOT NULL ,
          yom INT  NOT NULL
);
```

Hint: You can copy plain text from the clipboard into the MYSQL Monitor command prompt by right clicking. Make sure you have added a semicolon ";" at the end of the command.
Pressing ↵ will execute the command.

In **'MySQL Monitor'**, if you key in the query, rather than copy and paste, you will need to press 'enter key' ↵ after each line. This will generate the "->" characters that indicate that the current MySQL line is a continuation to the above line **not** a new MsSQL line. These lines subsequently form a single MySQL query that is terminated by a semicolon ";". Pressing ↵ after the semicolon will then execute the query.



Alternatively, in **phpMyAdmin**, you could create a table from the phpMyAdmin interface by selecting your database and then clicking the 'Create Table' tab, and then filling in the blanks OR enter the MySQL query by selecting the 'SQL' tab, typing or copying the SQL query, and clicking 'Go'.



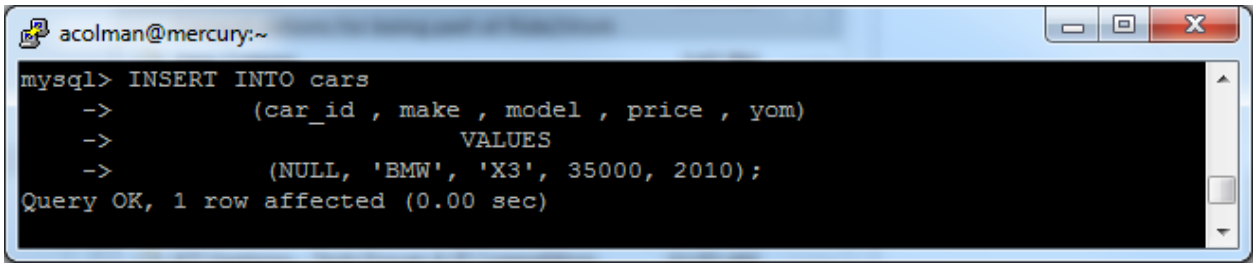
Task 4: Adding data

Records can also be added (inserted) using either the 'MySQL Monitor' command line or the phpMyAdmin interface.

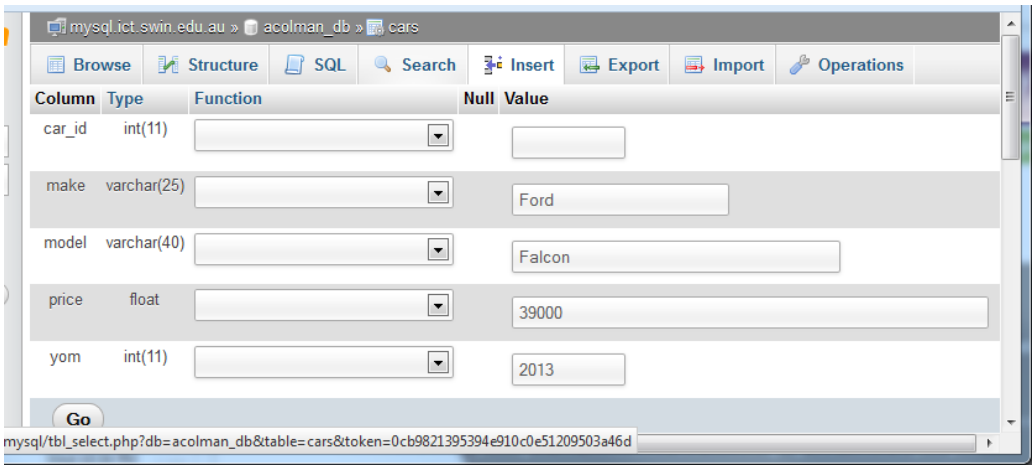
The SQL query will be:

```
INSERT INTO cars
      (car_id , make , model , price , yom)
            VALUES
      (NULL , 'BMW', 'X3', '35000', '2010');
```

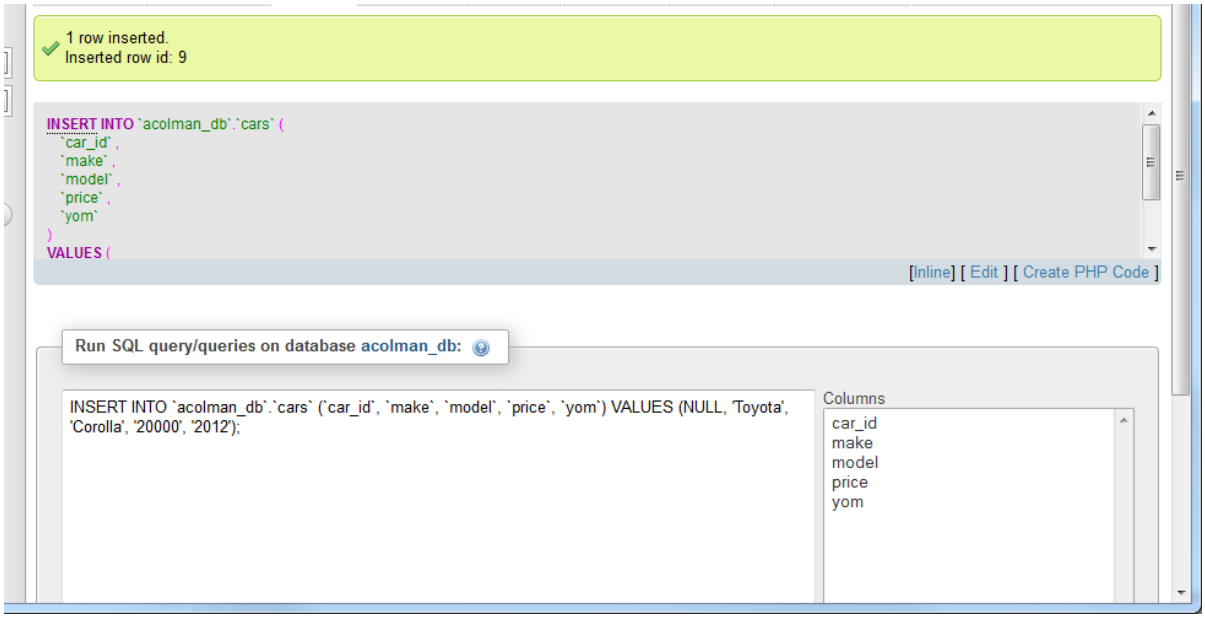
Note: single quotes can also be used around numbers.



Alternatively you can use the phpMyAdmin interface, and type in the details for a new record. As the car_id is auto-increment, we don't have to specify it.



The interface will then display the result along with the MySQL query it used to add the record:



Enter at least 10 records into the table using either the 'MySQL Monitor' command line or the phpMyAdmin interface. Sample data is shown in the table below.

Make	Model	Price	Year of Manufacture
Holden	Astra	$14,000.00	2009
BMW	X3	$35,000.00	2010
Ford	Falcon	$39,000.00	2013
Toyota	Corolla	$20,000.00	2012
Holden	Commodore	$13,500.00	2005
Holden	Astra	$8,000.00	2004
Holden	Commodore	$28,000.00	2009
Ford	Falcon	$14,000.00	2011
Ford	Falcon	$7,000.00	2003
Ford	Laser	$10,000.00	2010

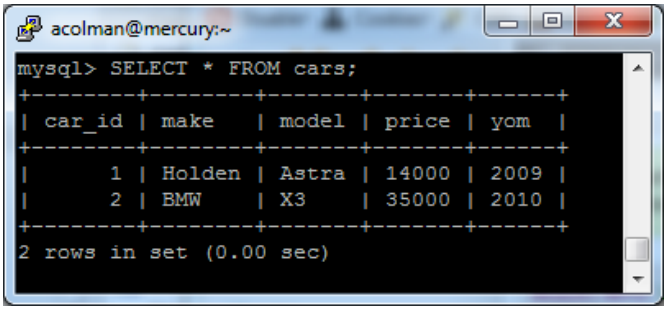
Task 5: Querying the table

In this task you will write queries that return records from the table.
In steps 1 to 4 below, the SQL is given. *Use SQL commands* to enter and send the following SQL queries to the database:

1. To select all records type:

   SELECT * FROM cars;

   NOTE: The executed SQL query will display the dataset requested.



2. To select make, model, and price, sorted by make and model

```
SELECT make, model, price FROM cars
            ORDER BY make, model;
```

3. To select make and model of the cars which cost $20,000.00 or more.

```
SELECT make, model FROM cars
           WHERE price >= 20000;
```

Note that when entering numbers, i.e. price, you must not specify the unit i.e. $ or comma.

4. The average price of cars for similar make.

```
SELECT AVG(price) FROM cars
            GROUP BY make;
```

Now write SQL for the following and the test the queries:

5. The make and model of the cars which cost below $15,000.00.

……

6. Average price of FORD cars.

……

7. Cars manufactured in 2010 or later and costing $15,000 or more.

……