WAD LABORATORY 4

1 Objectives covered in this laboratory

- To develop an understanding of the basic use of MySQL database
- To develop an understanding of the basic use of manipulating MySQL database tables using PHP.

2 Exercises

NOTE: Questions (a) to (e) are required exercises for all students (2.5pt each), while Questions (f) and (g) are additional exercises.

a) 1) Create a MySQL database table named 'inventory' in MySQL Monitor, a command-line program. The structure of inventory table is the shown in the lecture slides for Week 4 (slide #24).

2) Insert at least 5 records into the table.

3) Write a query that return all records of the table.

4) Update an existing table row using 'update' statement.

(You can find a step-by-step example of using MySQL database "A step-by-step example of using MySQL.pdf" in the Lab 4 folder downloaded from Canvas)

b) You may find it easier to use the MySQL web interface called phpMyAdmin which is available at

https://feenix-mariadb-web.swin.edu.au/

Create a MySQL database table named 'Employees' in phpMyAdmin. The structure of Employees table is the shown in the lecture slides for Week 4 (slide #3). Then, similar to a), use this web interface to insert some data into this table and write SQL codes to query/update the table.

c) Write a PHP page to retrieve records from the 'inventory' table created in a), and display them neatly in an html table. You need to provide an html select control on the page (see Figure 1). The content of this select control are names of all makes that can be found in the inventory table. User can select to show the data of a specific make or show the data of all makes (see Figure 2).

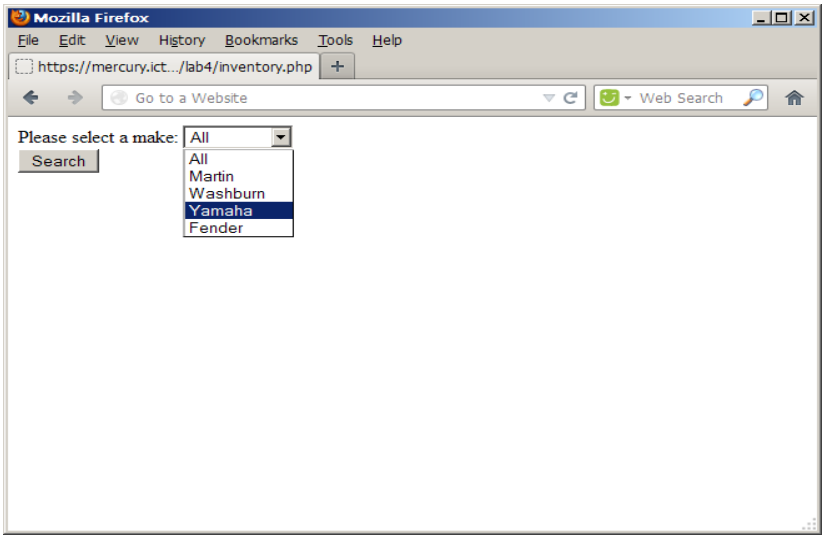


Figure 1 the content of html select control

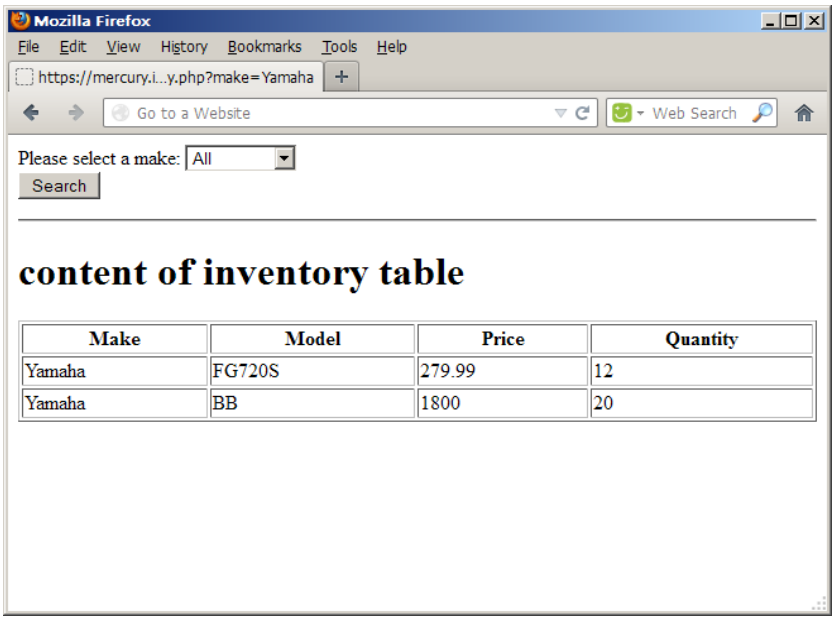


Figure 2 the item whose make is 'Yamaha'

d) Modify the page in c) to allow user insert new data row from the web page (see Figure 3). After the user input the content of new row and press the 'add' button, the input data will be inserted into MySQL database. Then the page will display the updated content of the 'inventory' table (see Figure 4).

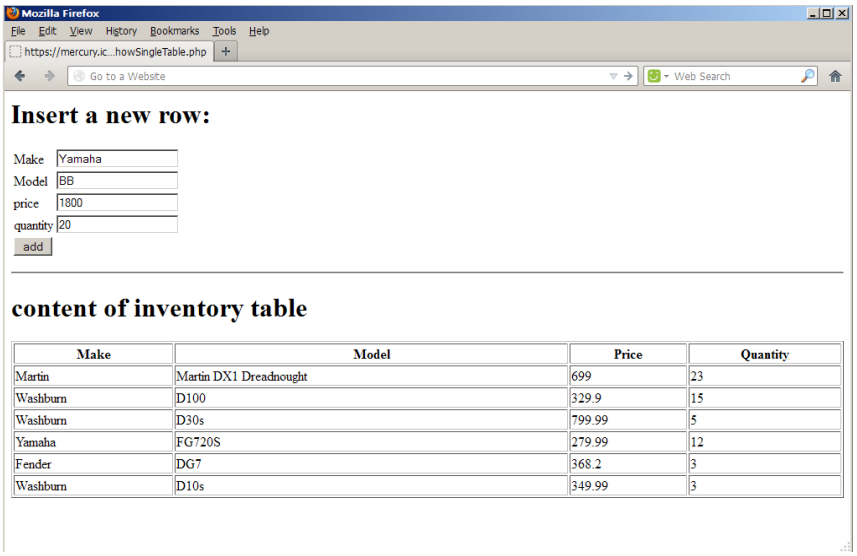


Figure 3 input data on the web page

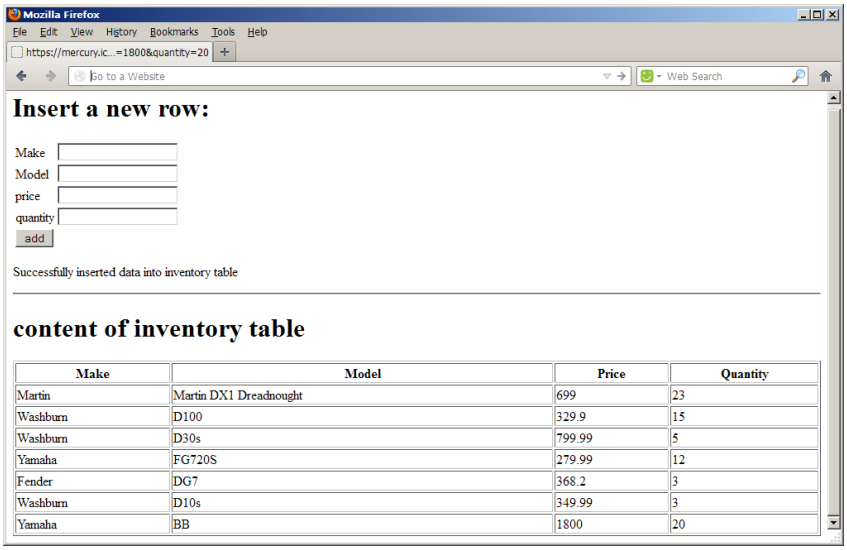


Figure 4 new row added

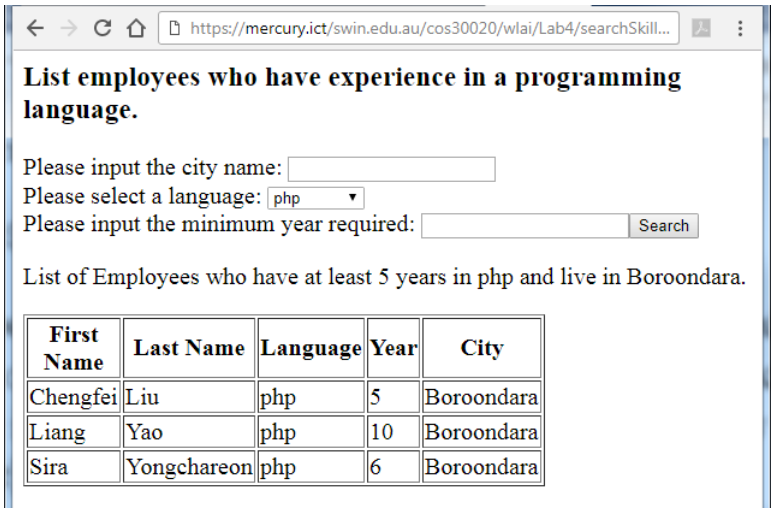
e) The following three tables are created with some data in a database called person_db.

```
Employees(employee_id, last_name, first_name, address,
city, state, zip);
Experience(employee_id, language_id, years);
Languages(language_id, language);
```

Write the SQL query statement to retrieve those employees (with their items: first_name, last_name, language, years, city) who have 5 years' experience in "PHP" and live in the city "Melbourne".

```
$sqlQuery = "SELECT ……
```

f) Modify the code `SearchSkill.php` introduced in Lecture 4 (download `Lec4Examples.zip` from Blackboard) to get city, language, and years from the interface and output the search result as below by using the similar SQL query in e):



(Refer to the lecture slides for Week 4: slides #58-62)

g) Change the simple online quiz PHP page that you wrote for exercise (e) in Lab2 and exercise (c) in Lab3. Instead of hard coding the questions on the HTML page or store them in a file, you need to store them in one or more MySQL database tables. You need to design the table structure to store questions and answers properly. When the page is loaded, it will read the questions from the database and display them on the screen. So if questions/answers are changed, this program can be used without modifying any of PHP codes.