

CNT 4714 – Project Five – Fall 2015

Title: “Project 5: Developing A Three-Tier Distributed Web-Based Application Using PHP and Apache”

Points: 100 points (115 points with the bonus problem completed)

Due Date: Sunday November 29, 2015 by 11:59 pm (WebCourses time)

Objectives: To incorporate many of the techniques you’ve learned this semester into a distributed web-based application which uses PHP running on an Apache server to access and maintain a persistent MySQL database.

Description: In this project you will utilize a suppliers/parts/jobs/shipments database similar to the one that was developed for Project 4 (we’ll use a new script and create a different version of the database for this project) and access this database through a client front-end. The first-tier (client-level front-end) of your application will be a simple HTML page that allows the client to enter SQL commands into a window and submit them to the database for processing. Note that the client front-end must provide for the login of the user so that any of our clients (see below) can login to the database to execute operations. The client front-end will provide the user the option of querying the database or updating the database (see Projects 3, and 4). The client front-end will run on any web-based browser that you would like to use. The second-tier will use PHP technology to extract values from the client-side form and submit the SQL command to the MySQL database server for processing. The results returned by the MySQL database will be formatted by PHP into an HTML document to be displayed in the user’s browser. We’ll implement the same business logic in this assignment that we used in Project 4 and increment the status of any supplier by 5 anytime any shipment in which the quantity is greater than or equal to 100 is entered into the database through insertion or update. Note that any update of quantity ≥ 100 will affect any supplier involved in a shipment with a quantity ≥ 100 . The example screen shots illustrate this case. An insert of a shipment tuple (S5, P6, J7, 400) will cause the status of every supplier who has a shipment with a quantity of 100 or greater to be increased by 5. In other words, even if a supplier’s shipment is not directly affected by the update, their status will be affected if they have any shipment with quantity ≥ 100 . (You can also employ the business logic used in the bonus problem of Project 4 if you choose for a 10 point bonus for this project.) The business logic of the second tier will reside on the Apache web-application server (server-side application in PHP). The third-tier (back-end) is the persistent MySQL database and is under control of the MySQL server.

We will assume that the front-end will allow any client to connect to the database, so like Project 3, you must provide the user the ability to enter login credentials on the initial screen (see screen shots below). Once a user has logged into the system, your main page should show the name of the currently logged in user and provide them the ability to logout (disconnect). The initial screen (application home page, should explain a bit about the application and explain the user login procedure. Once a user has logged into the system, the main application page should appear which provides the user the ability to issue SQL commands against the database. While your application screens do not need to look exactly like mine, they must provide the same functionality including the ability to submit both queries and updates, clear result windows, and login and logout of the system.

References:

Notes: Lecture Notes for MySQL Documentation for MySQL available at: <http://www.mysql.com>.
Lecture Notes for PHP. Projects 3, and 4.

Restrictions:

Your source file shall begin with comments containing the following information:

/* Name:

Course: CNT 4714 – Fall 2015

Assignment title: A Three-Tier Distributed Web-Based Application Using PHP and Apache

Due Date: November 29, 2015

***/**

Input Specification: The database for this project is created by running the script titled: `prog5dbscript`, which is available on WebCourses. Note that this is a new database and not the same one that was developed in project 4. The script creates a database named **project5**. All other input comes from the front-end client form submitted to the application server and processed by PHP entered as either queries or updates to this database.

The **first step** in this assignment is to login as the root user and execute/run the script to create and populate the backend database. This script is available on the course assignment page and is named “`prog5dbscript.sql`”. This script creates a database named **project5**. You can use the MySQL Workbench for this step, or the command line whichever you prefer.

The **second step** is to create authorizations for two client users (in addition to the root user) named **client1** and **client2**. By default your root user has all permissions on the **project5** database. Use either SQL Grant statements from the command line or the MySQL Workbench (see separate document for details on how to accomplish this task) to check and set permissions for the clients as follows:

Register the new user named **client1** (assign them an easy to remember password – ignore the MySQL warning on weak password setting) and assign to this user only selection and insertion privileges on the **project5** schema. Next, register the new user named **client2** and assign to this user only selection and update privileges on the **project5** schema.

Output Specification: All output is generated by the MySQL server, formatted by the PHP server-side application and should be processed into an HTML table or text (error pages) and returned by PHP to the user’s browser as a text/html page. Output will be generated for three different users, the root user, **client1**, and **client2**.

Deliverables:

- (1) Submit a working copy of your source code via WebCourses no later than **11:59pm Sunday November 29, 2015**. Include all of files that you generated to handle this application. I did my version of this project using one script file, but there are many different ways you could organize things, just be sure to include all your relevant files. Load all of your files into a folder named **Project5** that can be dropped into the Apache htdocs folder for testing.
- (2) Screen shots showing the output for each user. There will be a total of **31** different screen shots. Divide them up into three different files, one file for the root user (11 screen shots), one file for **client1** (10 screen shots), and one file for **client2** (10 screen shots). Screen shots should be generated in the following order: (1) create (re-create) the **project5** database, (2) root user commands, (3) **client1** commands, finally (4) **client2** commands.

Additional Information:

Remember that your PHP scripts belong in the `htdocs` directory under the Apache server. See the course notes on setting up Apache and PHP for the details.

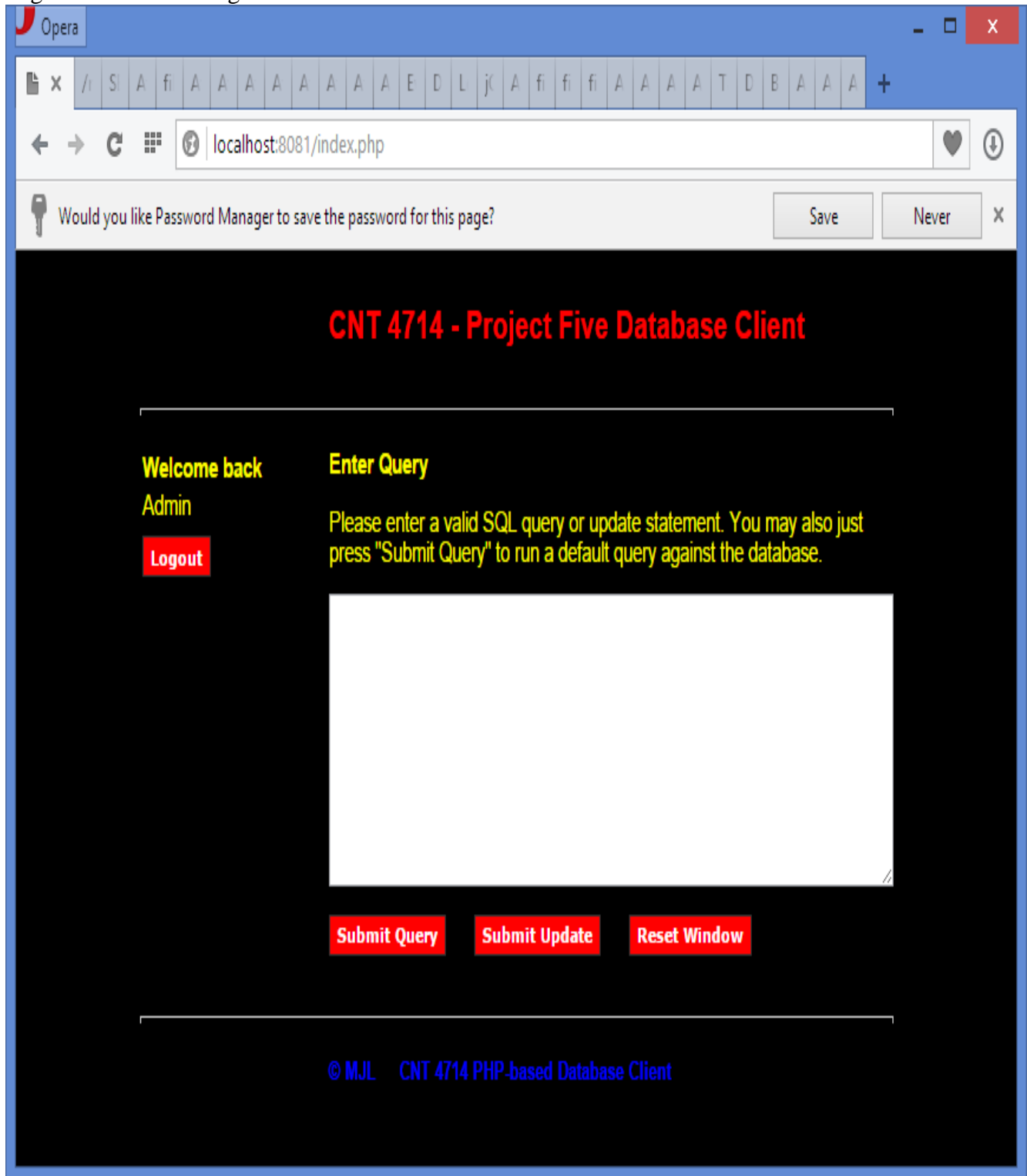
Some screen shots illustrating the application are shown on the next few pages. Except for the user interface, the output screen shots of this application will look pretty similar to that from Project 4 so I did not repeat all of the variances that I illustrated in that assignment document. You may want to look over Project 4 again if you've forgotten what some of the output should look like.

Screen Shots Illustrating the Application:

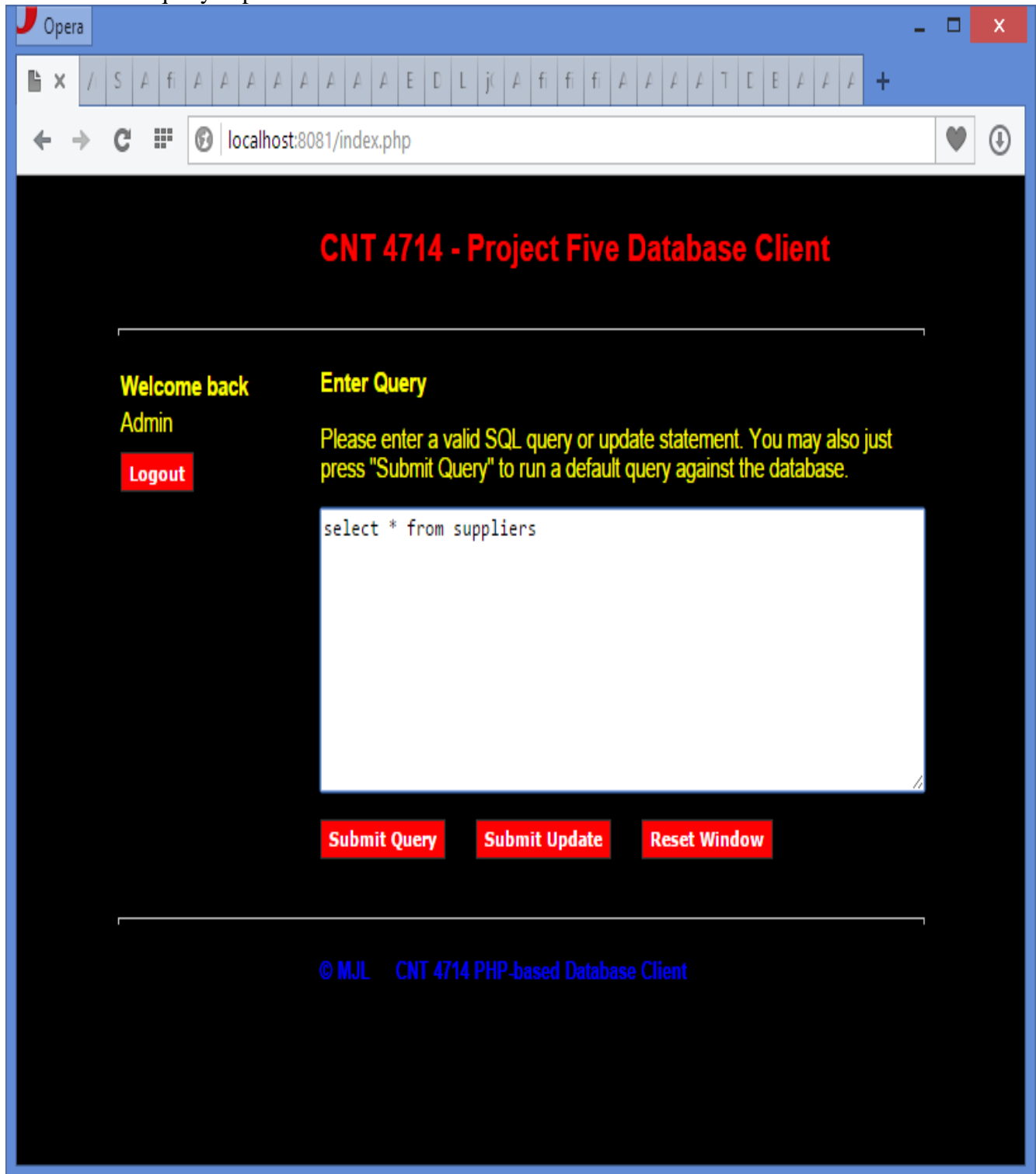
Main client login screen (initial application page):



Page after the client logs in:



Client enters a query expression:



Display of query results to the client:



CNT 4714 - Project Five Database Client

Welcome back
Admin
Logout

Query Results

snum	sname	status	city
S1	Michael Scuhmacher	1	Berlin
S10	David Coulthard	2	London
S11	Bernard Hinault	7	Paris
S12	Eddy Merckx	1	Brussels
S17	Rubens Barichello	3	Sao Paulo
S2	Juan Pablo Montoya	4	Interlagos
S22	Jan Ullrich	5	Bonn
S3	Dietrich Thurau	1	Berlin
S32	Bernd Schnieder	2	Berlin
S4	Mark Webber	0	Melbourne
S5	Jenson Button	4	London
S6	Nicola Gianniberti	2	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Rikonnen	2	Helsinki

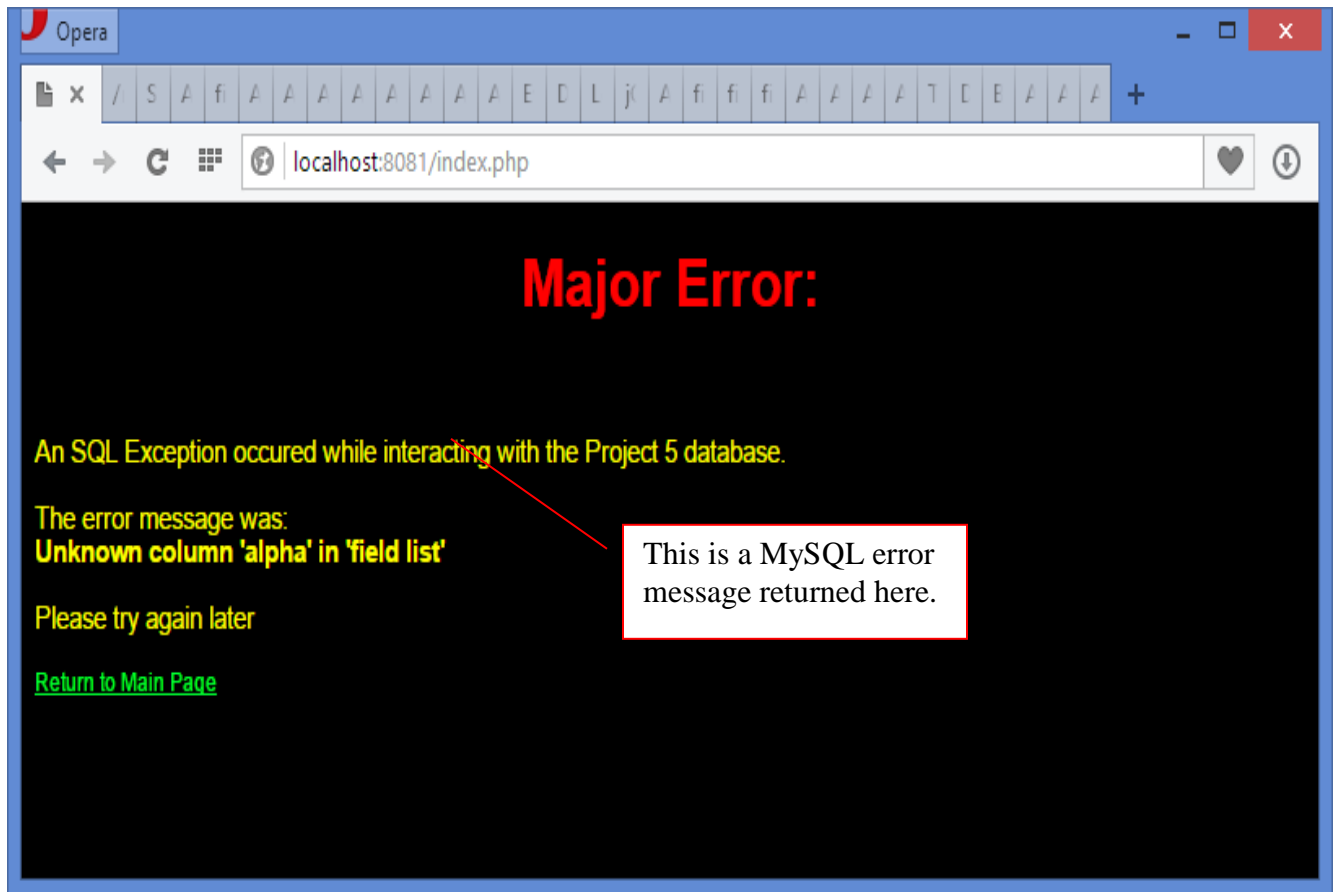
[Return to Main Page](#)

© MJL CNT 4714 PHP-based Database Client

Client enters an invalid query:



Error reporting screen illustrating return of MySQL error message to client:



Initial Suppliers table:

Opera

localhost:8081/index.php

CNT 4714 - Project Five Database Client

Welcome back
Admin
[Logout](#)

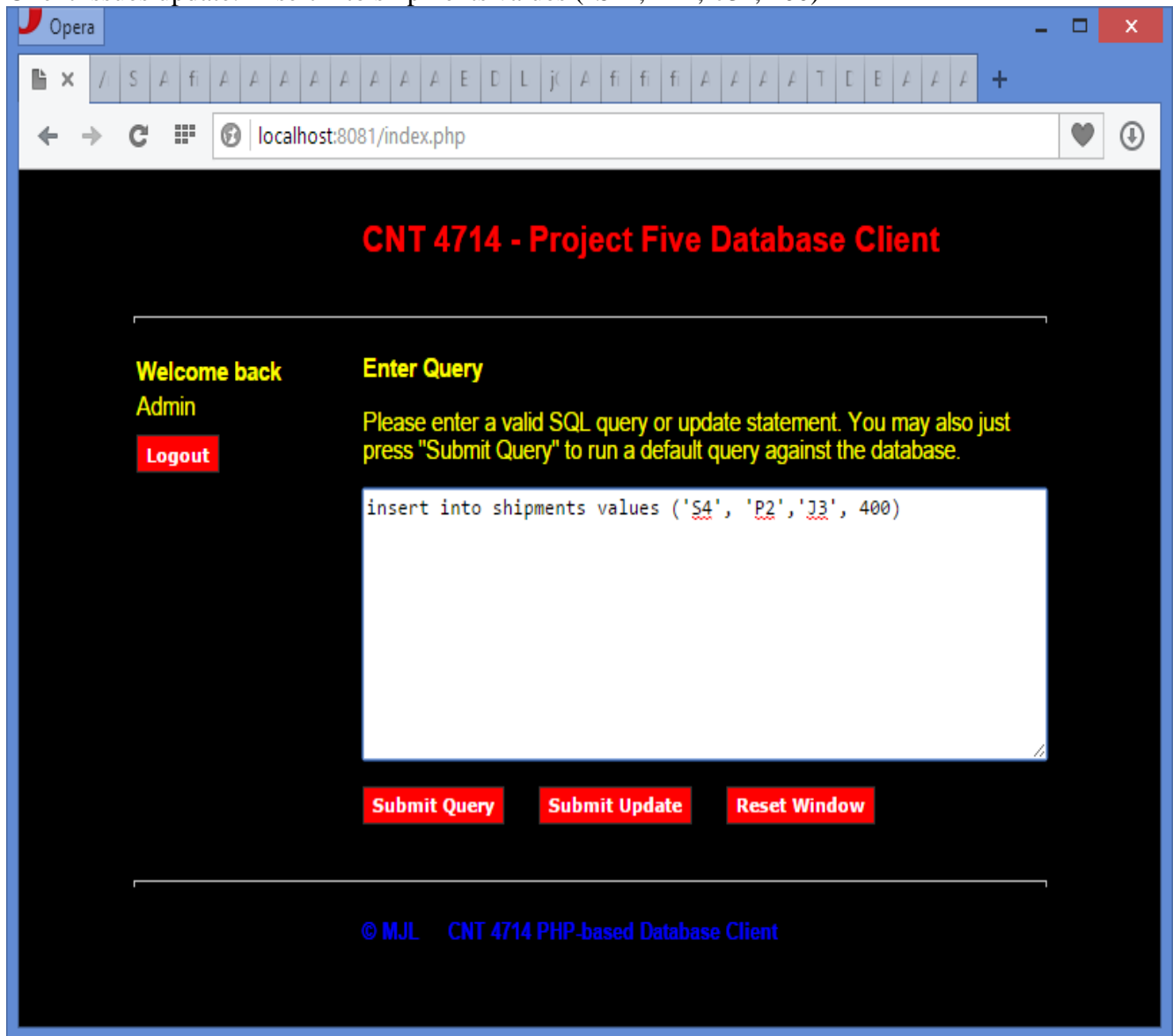
Query Results

snum	sname	status	city
S1	Michael Schumacher	1	Berlin
S10	David Coulthard	2	London
S11	Bernard Hinault	7	Paris
S12	Eddy Merckx	1	Brussels
S17	Rubens Barichello	3	Sao Paulo
S2	Juan Pablo Montoya	4	Interlagos
S22	Jan Ullrich	5	Bonn
S3	Dietrich Thurau	1	Berlin
S32	Bernd Schnieder	2	Berlin
S4	Mark Webber	0	Melbourne
S5	Jenson Button	4	London
S6	Nicola Gianniberti	2	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Rikonnen	2	Helsinki

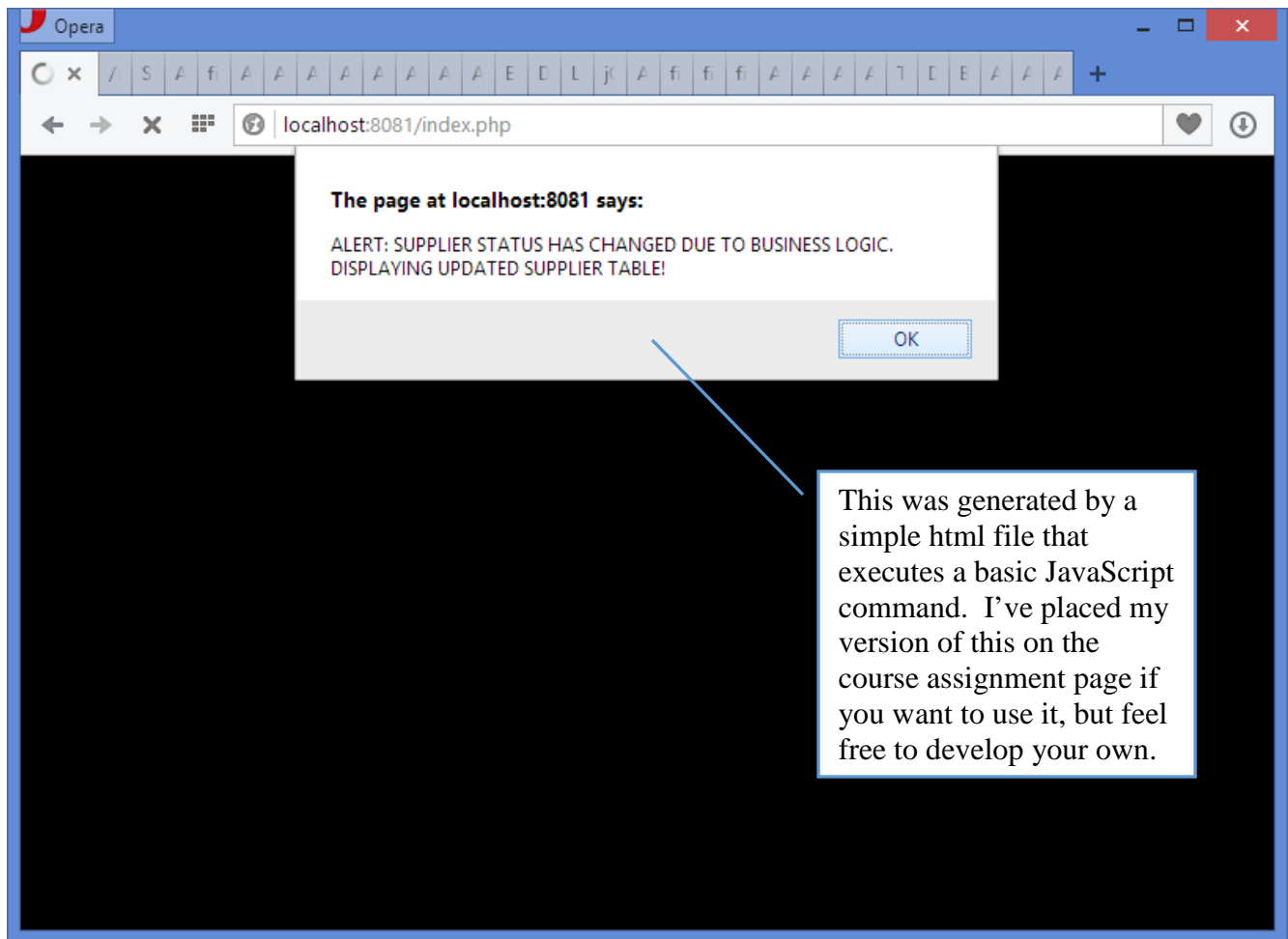
[Return to Main Page](#)

© MJL CNT 4714 PHP-based Database Client

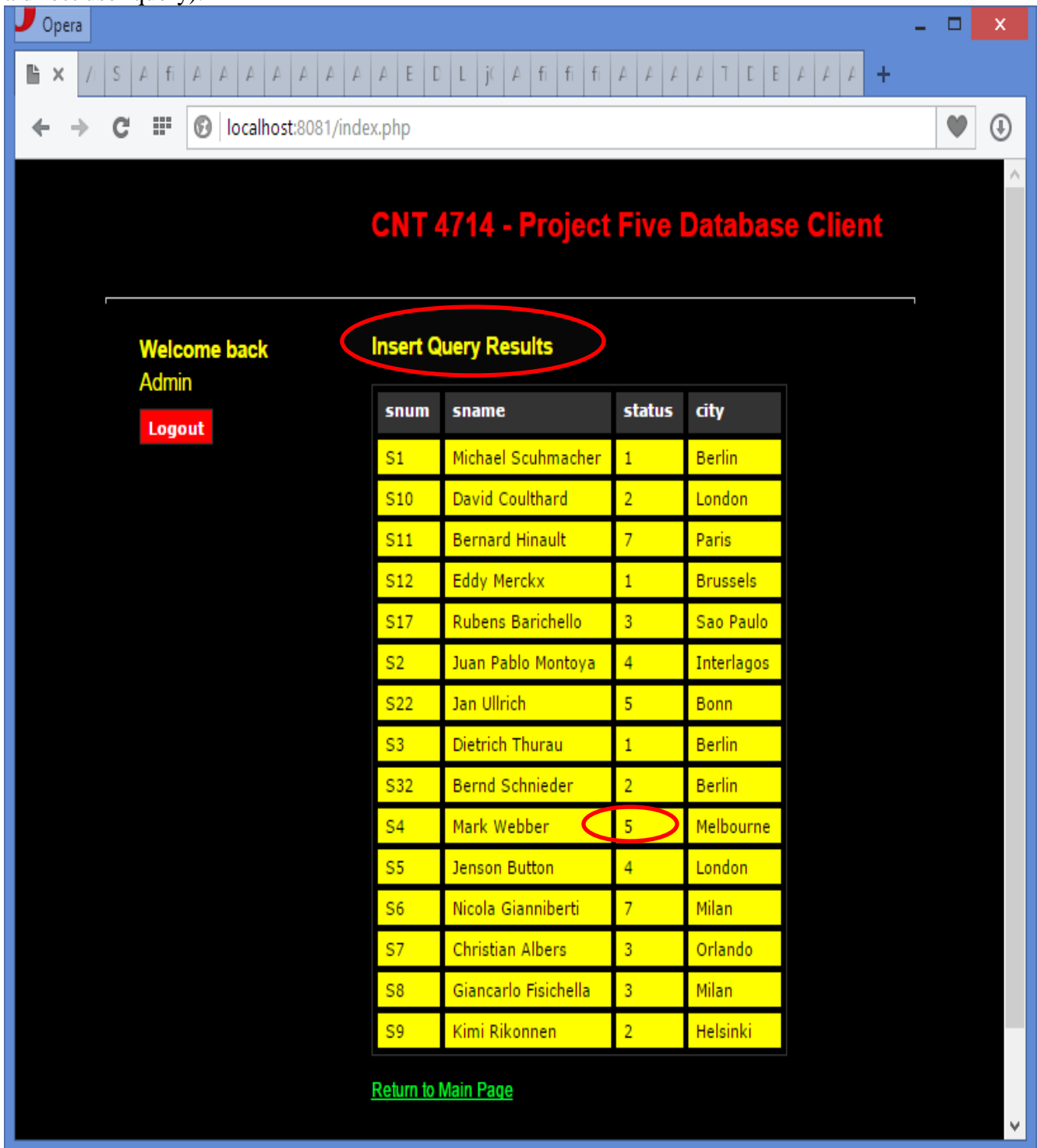
Client issues update: insert into shipments values ("S4","P2","J3", 400)



Client is notified that business logic was activated:



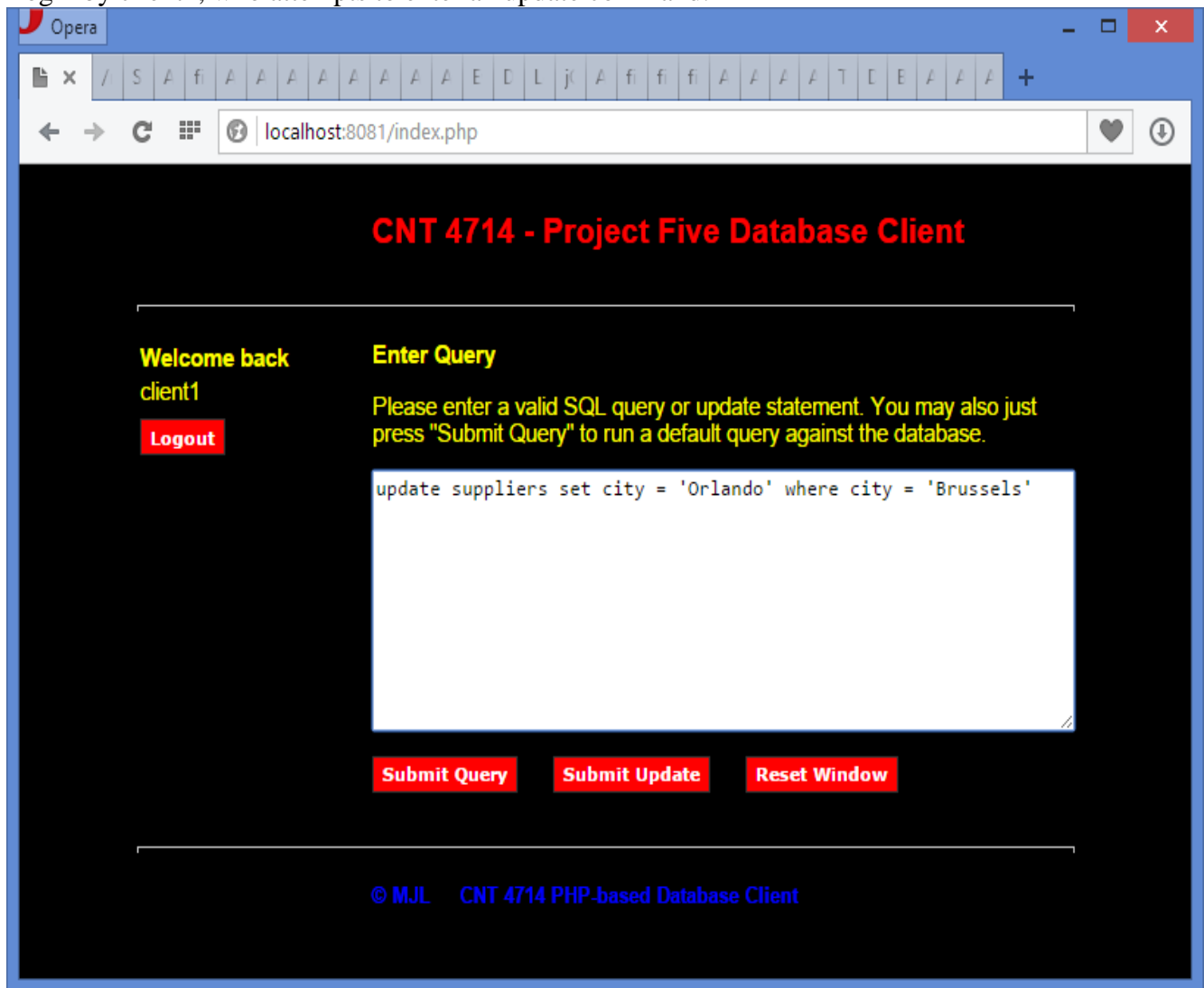
Automatically generated query of “SELECT * FROM suppliers” to illustrate the effects of the business logic being triggered by the previous update (note that this is distinguished from a normal query by the header that indicates this is an insert query result – meaning triggered by the business logic rather than a direct user query):



The screenshot shows a web browser window with the address bar displaying 'localhost:8081/index.php'. The page title is 'CNT 4714 - Project Five Database Client'. On the left side, there is a sidebar with the text 'Welcome back Admin' and a red 'Logout' button. The main content area features a section titled 'Insert Query Results' which is circled in red. Below this title is a table with four columns: 'snum', 'sname', 'status', and 'city'. The table contains 16 rows of data. The row for 'Mark Webber' with status '5' is circled in red. At the bottom of the page, there is a green link that says 'Return to Main Page'.

snum	sname	status	city
S1	Michael Scuhmacher	1	Berlin
S10	David Coulthard	2	London
S11	Bernard Hinault	7	Paris
S12	Eddy Merckx	1	Brussels
S17	Rubens Barichello	3	Sao Paulo
S2	Juan Pablo Montoya	4	Interlagos
S22	Jan Ullrich	5	Bonn
S3	Dietrich Thurau	1	Berlin
S32	Bernd Schnieder	2	Berlin
S4	Mark Webber	5	Melbourne
S5	Jenson Button	4	London
S6	Nicola Gianniberti	7	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Rikonnen	2	Helsinki

Login by client1, who attempts to enter an update command:



MySQL returns the error that you saw in the Workbench version when a privilege was not granted (see last page).

