CS 377 MySQL/PHP Tutorial

Connect from PHP Scripts

MySQL is currently the most popular open source database server in existence. On top of that, it is very commonly used in conjunction with PHP scripts to create powerful and dynamic server-side applications.

mysql connect

Before we can do anything with MySQL in PHP we must first establish a connection to the web host's MySQL database. This is done with the MySQL connect function.

Example 1:

```
<?php
mysql_connect('compute.mathcs.emory.edu:8228/tmp/mysql.cs377', 'lab2', 'student') or
die(mysql_error());
echo "Connected to MySQL<br />";
?>
```

If you load the above PHP script to the webserver (i.e., save the code in a .php file, say, test.php, under your *~/cs377/web/* directory) and everything works properly, then you should see "Connected to MySQL" displayed when you view the test.php page via:
http://cs377.mathcs.emory.edu/~yourid/test.php

The ***mysql_connect*** function takes three arguments. Server, username, and password. In our example above these arguments were:

Server - compute.mathcs.emory.edu:8228/tmp/mysql.cs377
Username - lab2
Password - student
The "or die(mysql..." code displays an error message in the browser if -- you've probably guessed it -- there is an error in processing the connection. Double-check your username, password, or server if you receive this error.

Note that ‘echo’ is used to print messages in PHP.

mysql_query / my_fetch_array

Once a database is connected, various things can be done, including creating tables, inserting, selecting and updating rows in a table. All of these can be done through SQL queries, which can be executed via invoking the ***mysql_query*** function. In addition, ***my_fetch_array*** function can be used to read a row from the result set returned by ***mysql_query***.

Example 2:

```
<?php
// Make a MySQL Connection
$result = mysql_query('select * from cs377.employee');

while ($row = mysql_fetch_array($result)) {
      echo "fname = " . $row['fname'] .
       ", lname = " . $row['lname'] . "<br>";
   }
?>
```

The value that ***mysql_query*** returns and stores into ***$result*** is a special type of data, it is a MySQL Resource. The ***mysql_fetch_array*** function takes a MySQL query resource as an argument ($result) and returns the first row of data returned by the mysql_query. The columns of the MySQL Result can be accessed by using the column names of the table. In our example these are: ***fname*** and ***lname***. The ***mysql_fetch_array*** function also returns ***FALSE*** if there are no more rows to return. Using a PHP While Loop we can use this information to our advantage. As shown in the example, placing the statement "$row = mysql_fetch_array()" as our while loop's conditional statement will accomplish two things:

- We will get a new row of MySQL information that we can print out each time the while loop checks its conditional statement.
- When there are no more rows the function will return FALSE causing the while loop to stop.

Note that '.' is used to connect strings in PHP.

mysql_close

mysql_close() closes connection to the MySQL server. However, using ***mysql_close()*** isn't usually necessary, as non-persistent open links are automatically closed at the end of the script's execution.

Example 3:

```
<?php
$link = mysql_connect('compute.mathcs.emory.edu:8228/tmp/mysql.cs377', 'lab2', 'student') ;
// Execute some SQL Queries
mysql_close($link);
?>
```

Additional References

http://www.php.net/manual/en/ref.mysql.php

http://www.php.net/manual/en/getting-started.php

http://www.tizag.com/mysqlTutorial/

Direct Connect from a Console

MySQL can also be connected directly from a console as follows.

1. To log in, you can type in the following in a console:

```
mysql -u username -p --sock=/tmp/mysql.cs377.sock
```

and then type in your password.

2. Once you login to MySQL, you can check which databases are available for you to use, by typing the following:

```
show databases;
```

and you will see something like below.

```
+----------+
| Database|
+----------+
| cs377    |
| grpx     |
| test     |
+----------+
```

3. Then you can select the database of interest (say, “cs377”) by typing:

```
use cs377 ;
```

4. To see which tables are in the selected database, you can type in:

```
show tables;
```

you will see something like below:

```
+----------------------+
| Tables_in_cs377      |
+----------------------+
| department           |
| dependent            |
| dept_loc             |
| employee             |
| example_autoincrement |
| project              |
| works_on             |
+----------------------+
```

5. If you are interested in the schema of a table (say, “department”), you can type in:

```
desc department;
```

you will see something like below:

```
+--------------+----------+------+-----+---------+-------+
| Field        | Type     | Null | Key | Default | Extra |
+--------------+----------+------+-----+---------+-------+
| dname        | char(15) |      |     |         |       |
| dnumber      | int(11)  |      |     | 0       |       |
| mgrssn       | char(9)  |      |     |         |       |
| mgrstartdate | char(10) | YES  |     | NULL    |       |
+--------------+----------+------+-----+---------+-------+
```

6. Now you can submit different kinds of SQL queries according to your need. Additional details and examples can be found in the following MySQL manual online.

Additional References

http://dev.mysql.com/doc/refman/4.1/en/index.html