Lab 10 – PHP and MySQL

Note that documentation is available online here:
http://www.php.net/manual/en/ref.mysql.php Basically, there are four things you want to be able to do in MySQL from within PHP:

1. Connect to MySQL DB
2. Execute MySQL queries
3. Check the status of your MYSQL commands

Queries can be any kind of MySQL query, including SELECT, UPDATE, INSERT, etc. Following SELECT queries, you can execute MySQL/PHP functions to put the data read from the MySQL database into PHP variables. Then you can use the PHP variables in your PHP script to do whatever analysis, display, etc. that you want.

1. Connect to MySQL Database

Here is an example of connecting to the MySQL database from within PHP:

```
$conn=mysql_connect($mysql_host,$mysql_user,$mysql_password) or die('Could
not connect: '.mysql_error());
echo 'Connected successfully';
mysql_select_db( $mysql_db ) or die( 'Could not select database' );
```

Notice that there are two functions invoked:
- one that logs into mysql: mysql_connect()
- one that selects the database to use: mysql_select_db()

2. Execute MySQL Queries:

Here is an example of executing a SELECT query from within PHP:

```
// set up and execute the MySQL query
$query = 'SELECT * FROM my_table';
$result = mysql_query( $query ) or die( 'Query failed: '. mysql_error() );
// print the results as an HTML table
echo "<table>\n";
while ( $row = mysql_fetch_array( $result, MYSQL_ASSOC )) {
echo "\t<tr>\n";
foreach ( $row as $item ) {
echo "\t\t<td>$item</td>\n";
}
echo "\t</tr>\n";
}
echo "</table>\n";
// free result
mysql_free_result( $result );
```

There are two functions used here:

- one to execute the query and store the result in a local variable: mysql_query()
- one to parse the data read returned from the query as an array: mysql_fetch_array()

3. Check the status of your MySQL commands

If errors occur, the functions return errors. These errors can be read as strings using the function mysql_error().

Note the usage in this statement:

```
$conn=mysql_connect($mysql_host,$mysql_user,$mysql_password) or die('Could
not connect: '.mysql_error());
echo 'Connected successfully';
```

B. Try things out for yourself.

Create a database with the name of your own choice. In it create a table that will hold your basic information.

1. Write a PHP script that will connect to your MySQL database and query your created table, selecting all the rows and columns. Display the results as an HTML table in a web page.
2. Modify the query to select only the 'thename' column, for all rows in your birthday table.
3. Modify the query to select all the columns in your created table for those rows with a birth month in March.
4. Modify the query to select only the 'thename' column in your birthday table for those rows with a birth month in March.