

# Practical Lab Exercises

## Lab – PHP & DB

Web Programming (F28WP)

---

### Introduction

In this lab, you'll further develop your understanding of web programming concepts

### 1.1 PHP and MySQL

Setup your MySQL database and write a very simple script in PHP to see if you can connect with the database.



Your MySQL username and password:

Your MySQL username is your usual HWU username ([USERNAME]).

For example sam77

Your MySQL password has this pattern: abc[USERNAME]354.

For example the password for sam77 will be abcsam77354

Therefore, when you are in the Linux lab and if you are sam77, you would enter the following command to make a connection to MySQL and access your database, which will be called sam77 too:

```
mysql -u sam77 -D sam77 -h mysql-server-1 -p
```

Enter password:

abcsam77354

You can also use <https://www.macs.hw.ac.uk/phpMyAdmin> to login to the MySQL server.

MySQL and PHP References:

- <http://www.mysqltutorial.org>
- [\*Beginning PHP and MySQL From Novice to Professional 4th Edition \(2010\) by W. Jason Gilmore.\*](#)

## 1.2 PHP and MySQL (public\_html)

This exercise, you'll create a simple php script which will access your MySQL database you configured using 'phpMyAdmin' in the previous section.

Task 1: Create the "users" table in your database by implementing the following steps:

1. Login to <https://www.macs.hw.ac.uk/phpMyAdmin>
2. Select your database
3. Open a SQL input box (to open an input box, select or click the "SQL" link on your screen)
4. Copy and Paste the below users-table SQL code into the input box (Copy all the text that is between "-- start here" and "-- end here").
5. Hit the "Go" button
6. Refresh the webpage of <https://www.macs.hw.ac.uk/phpMyAdmin> and check that your users table is included in your list of tables.

```
-- start here
-- users-table SQL code:
--
-- Table structure for table `users`
--
```

```
CREATE TABLE `users` (
  `userid` smallint(4) UNSIGNED NOT NULL,
  `username` varchar(20) NOT NULL,
  `userpassword` varchar(20) NOT NULL,
  `name` varchar(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
INSERT INTO `users` (`userid`, `username`, `userpassword`, `name`) VALUES
(1, 'peter', '123peterabc', 'Peter Smith'),
(2, 'mary', '123maryabc', 'Mary Poppins'),
(3, 'jimmy', '123jimmyabc', 'Jimmy Carter'),
(4, 'ada', '123adaabc', 'Ada Yonath');
```

```
ALTER TABLE `users`
  ADD PRIMARY KEY (`userid`);
```

```
ALTER TABLE `users`
  MODIFY `userid` smallint(4) UNSIGNED NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=5;
```

```
-- end here
```

Task 2: Create the myfirstscript.php in your public\_html directory:

To do this task you will need to remotely login to the MACS linux server. You can use your favourite "telnet" programme to remotely login to the server. Normally you can use Putty. You can download Putty from <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>. You only need the "the SSH and Telnet client" 32-bit version (putty.exe) . Run putty and enter jove.macs.hw.ac.uk as the hostname of the server. Make sure to enter 22 as the Port number (to establish a secure ssh connection). Hit "Open" to open a remote "putty screen". Now, enter your linux username and password to login to the MACS server. If your username/password doesn't work, contact [help@macs.hw.ac.uk](mailto:help@macs.hw.ac.uk).

Once you have logged in, do as follows:

1. Open your browser and create a .php file
2. See below for an example php script for connecting to the database
4. Go back to your "putty screen" and type: cd public\_html and hit "enter" to go to your web directory.
5. Type vi myfirstscript.php and hit "ENTER". This will open a blank page with the vi editor.
6. Hit the i letter to change your editor to the "INSERT" mode.
7. Right-click your mouse to paste the clipboard contents
8. Hit the "ESC" key to close the "INSERT" mode
9. Type :wq (all those three keys together, without spaces) and hit "ENTER". This will save your myfirstscript.php script.
10. Create a directory called uploads in your public\_html directory.
11. Run myfirstscript.php on your browser to upload a file to your new uploads directory.
12. If something doesn't work (e.g. you cannot upload files) remember how we fixed the errors during the demo done in the lecture of last Thursday.

You should be able to develop a php script which will read and write to the MySQL database.

The php script below provides a skeleton example, from which to start. Experiment by customizing the database, structure, writing and writing simple and more complex data sets so you are comfortable with working with PHP and MySQL.

```
<?php
//DATABASE CODE (Example):
/*
$db_connected = connectDB('yourDatabase');
$sql = "SELECT * FROM users WHERE username='".$$_POST["user"]."'" ;
$result=mysql_query($sql) or die($sql."<br>\n".mysql_error());
while($row = mysql_fetch_array($result)) {
    echo "Hello ";
    foreach($row as $col) echo " $col "; echo "<br>\n";
}
exit;
function connectDB($database='') {
    global $db, $mysqluser, $mysqlpwd;
    // initiate a database connection by giving a database name, username and
password:
    if($database=='') $database = 'yourDatabase';
    if($mysqluser=='') $mysqluser = 'yourUsername';
    if(!isset($mysqlpwd)) $mysqlpwd = "yourPassword";
    $db = new db_connection("mysql");
    if($db->connect("mysql-server-1.macs.hw.ac.uk", "", $mysqluser, $mysqlpwd,
0,$database)) return true;
    else return false;
}
class db_connection {
    var $connection;
    // create a new connection object
    function db_connection($type="") { }
    // connect to the database server
    function connect($host, $port, $login, $password, $pconnect, $database="") {
        if($port) { $host .= ":$port"; }
        if( !($this->connection = @mysql_connect($host, $login, $password)) )
return false;
        if($database) if(!@mysql_select_db($database, $this->connection)) return
false;
        return true;
    }
}
```

```
function query($query) {  
    return mysql_query($query, $this->connection);  
}  
function error() {  
    return mysql_error($this->connection);  
}  
}  
*/  
?>
```