

Assignment 2 — PHP and MySQL

TA: Patty (s221274155@gmail.com)

Deadline: 11:59pm, May 30

ATTENTION: You can freely use any PHP libraries and packages. However, you must implement all the functions required by this homework. If you fail to do that, you will get penalties.

1. Task Description

In this assignment, you will learn:

- How to manipulate data (text and image) by accessing MySQL and get more familiar with SQL statements
- How to use PHP and perform form processing and connect to a MySQL database
- How to manipulate sessions
- Get experience using JSON parsers in PHP and JavaScript

2. Before you start

Before you start with this assignment, you need to install Apache, PHP, and MySQL. You can install AppServ (<https://www.appserv.org/en/>), which is a package for all these three. For detailed installation guide, please check the official site for more information.

3. MySQL (10%)

- (1) Please connect to the MySQL server you installed
- (2) Design your own database schema and create tables. Here is an example for creating database tables with a primary key, index and a foreign key declaration. You will need to modify the SQL statements by yourself. You should replace mydb to your database.

```
USE mydb;
```

```
CREATE TABLE parent (id INT NOT NULL AUTO_INCREMENT, PRIMARY KEY (id));
```

```
CREATE TABLE child (id INT, parent_id INT, INDEX par_ind (parent_id), FOREIGN KEY↵  
(parent_id) REFERENCES parent(id));
```

- (3) Use PHP to connect to MySQL. Here is an example of how to use PHP to connect to MySQL. You need to change the database name, user name and password.

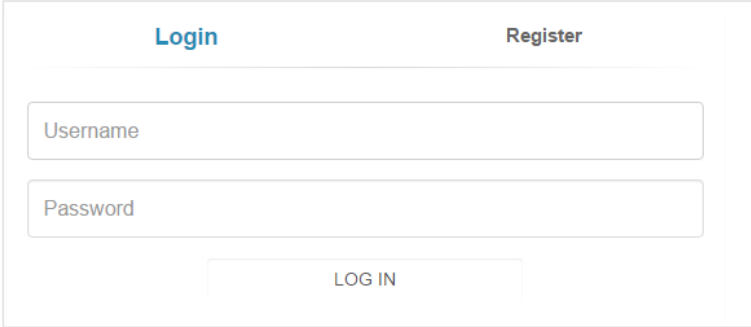
```
<?php
$link = mysql_connect('servername', 'your_mysql_username', 'your_mysql_password');
if(!$link){
    die("Could not connect: ", mysql_error());
}
echo 'Connected successfully';
mysql_close($link);
}
```

4. My Weather Implementation (85%)

Please implement a Weather website on which registered users can post **their thoughts**. Other registered users can comment and like. Additionally, the website must provide an interface to create an account. The functions in the website are described in the following sections.

4.1 Sign-in page

First, users sign in through this page. You must use a cookie to automatically fill the name of the user who has ever signed in before. If a user fails to sign in, please show an error page; otherwise, direct the page to your home page (see Figure 5). If a user clicks the “Register” link, he/she will be re-directed to the register page (see Figure 2). An example of a sign-in page is shown below in Figure 1.



The figure shows a sign-in page layout. At the top, there are two links: "Login" (in blue) and "Register". Below these links are two input fields: "Username" and "Password". At the bottom, there is a button labeled "LOG IN".

Figure 1: Example of a sign-in page

4.2 Register page

On this page, a user can register by entering a **name**, a **password**, and **confirm password**. An example is shown below in Figure 2.

Hit: The password in the database must be encrypted. Be aware of this when designing the database table schema.

The image shows a registration form with a light gray border. At the top, there are two links: "Login" and "Register". The "Register" link is highlighted in blue. Below the links, there are three input fields: "Username", "Password", and "Confirm Password". Each field has a light gray border and a small "x" icon on the right side. Below the input fields, there is a button labeled "REGISTER NOW" with a light gray border and a small "x" icon on the right side.

Figure 2: Example of a register page

4.3 Home page

First, you must obtain the weather open data in JSON or XML format by PHP. You can see the sample weather data in JSON and XML format in Figure 3 and Figure 4, respectively.

In the home page, you must rotate the weather information of a county (city) for a week for 5 seconds as illustrated in Figure 5 (a). You can obtain the weather information from one of the following two sites.

(1) Yahoo weather API: the weather information is updated every 2 to 3 hours.

<https://developer.yahoo.com/weather/>

(2) Meteorological data opening platform: the weather information is updated every 15 to 30 minutes.

Users can check the detailed local weather of the day as shown in Figure 5 (b), by clicking the dropdown menu in Figure 5 (a).

4.3.1 Parsing JSON-formatted data in PHP

In PHP 5, you can parse JSON-formatted data using the “json_decode” function. For more information, please go to <http://php.net/manual/en/function.json-decode.php>.

You can encode data into JSON-formatted objects using the “json_encode” function. For more information, please go to <http://php.net/manual/en/function.json-encode.php>.

4.3.2 Read and save contents in PHP

To read the contents of a JSON-formatted object, you can use the “file_get_contents” function.

To save contents on the server side, you can use “file_put_contents” function.



Figure 3: JSON page

```
<?xml version="1.0" encoding="UTF-8"?>
<ATM00698>
  <Data>
    <SiteName>南沙島</SiteName>
    <WindDirection>東北東</WindDirection>
    <WindPower>3</WindPower>
    <Gust/>
    <Visibility>10</Visibility>
    <Temperature>30.0(+0.1)</Temperature>
    <Moisture>79</Moisture>
    <AtmosphericPressure>1012.0</AtmosphericPressure>
    <Weather>多雲</Weather>
    <Rainfall1day>0.0</Rainfall1day>
    <Unit>中央氣象局</Unit>
    <DataCreationDate>107/3/1 11:00:00</DataCreationDate>
  </Data>
  <Data>
    <SiteName>東沙島</SiteName>
    <WindDirection>東北東</WindDirection>
    <WindPower>3</WindPower>
    <Gust/>
    <Visibility>10</Visibility>
    <Temperature>27.2(-0.2)</Temperature>
    <Moisture>83</Moisture>
    <AtmosphericPressure>1013.4</AtmosphericPressure>
    <Weather>晴</Weather>
    <Rainfall1day>0.0</Rainfall1day>
    <Unit>中央氣象局</Unit>
    <DataCreationDate>107/3/1 11:00:00</DataCreationDate>
  </Data>
</ATM00698>
```

Figure 4: XML page



Figure 5: A example of a home page

4.4 The message page and post page

On the message page, all posts and comments of all users are shown. The layout of the message page is shown in Figures 6 and 7. The newest post is shown on the top but the newest comment of a post is shown on the bottom. Users can like and unlike a post by clicking “like” link, and the number of likes is incremented by one. Once the user likes a post, he or she can unlike it later by clicking the “like” link again. Subsequently, the number of likes is deducted by one. The posts page will be updated when any post or comment is added. If a user accesses the message page without signing in, he/she will be redirected to the sign-in page. In addition, the header should have “Sign out” link. **You need to implement sessions to implement these features.**

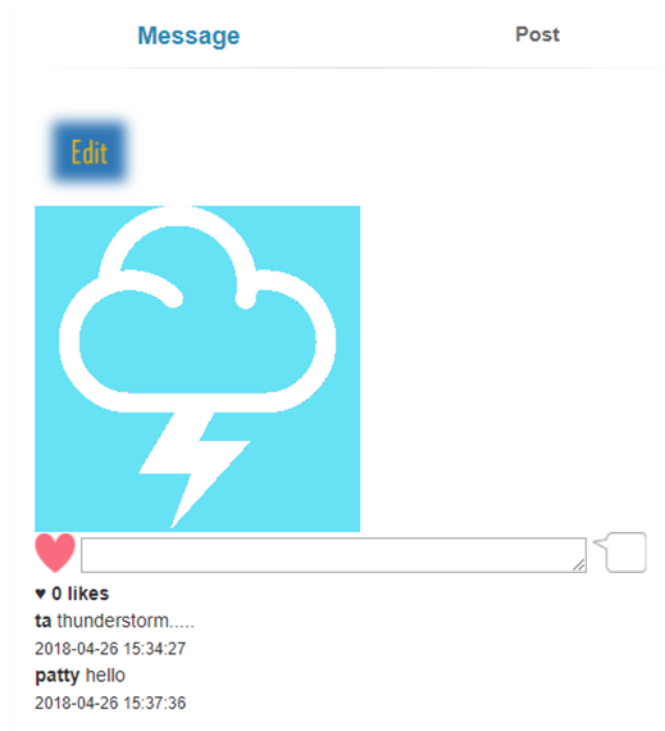


Figure 6: The message page.

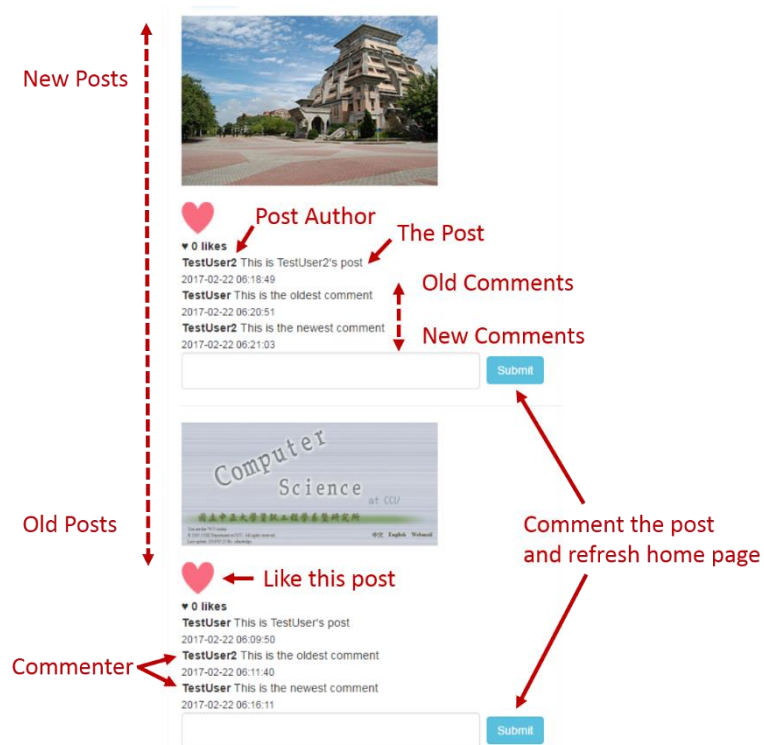


Figure 7: Explanation of the message page.

On the message page, there is a link to the post page (see Figure 8 (a)), where users can post a picture and their thoughts in text. Each post should contain a picture and texts at the same time; otherwise, an error message should be shown.

When clicking on “Upload” button, the picture will be uploaded into MySQL. When the picture is successfully uploaded, the user will be redirected to the next page (see Figure 8 (b)). The user can write his or her thoughts on “Write a post” area. Finally, the user can click on “Post” button, he/she will be redirected back to the message page.

ATTENTION: According to the MySQL restrictions, the image size should less than 85 KB. If the user uploads the image which size more than 85 KB, please show an error page.

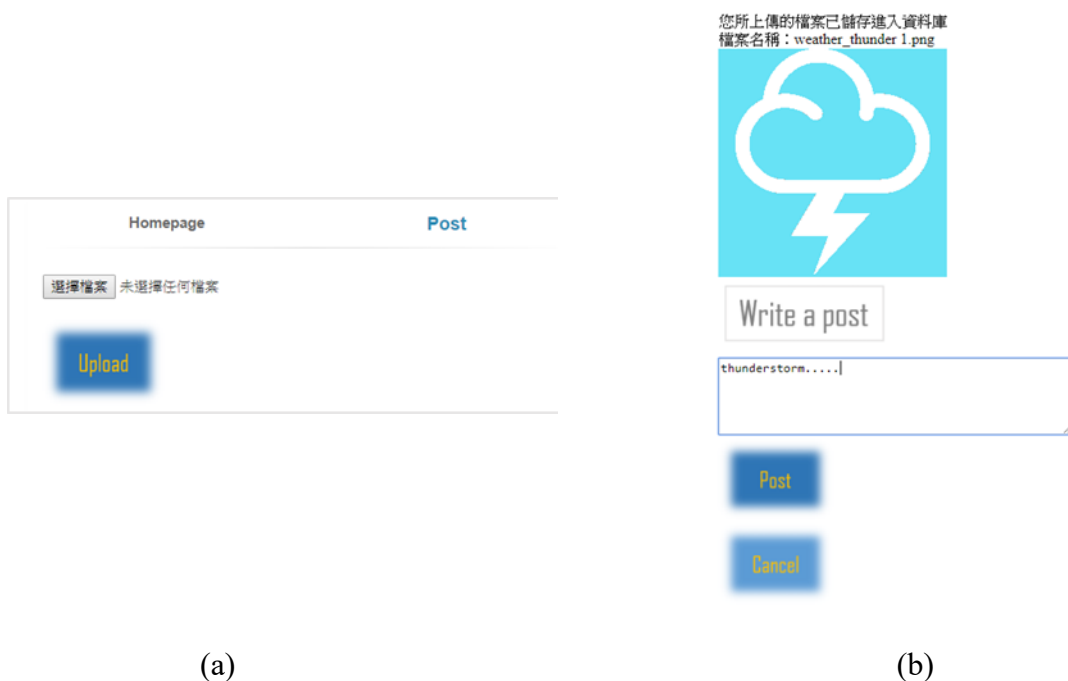


Figure 8: The post page

4.5 The introduction page

You need to use CSS to introduce this homework to style the introduction page as shown in Figure 9.



Figure 9: Introduction page

5. Additional features (Bonus)

You can freely create more features to your design. You must describe the details of features you add and provide the instructions to use them. The scores you will get is based on how functional and useful your advanced features are and the description you provide.

Here are some suggestions:

- Add a timestamp for each post and comment. (3 pts)
- The post can be edited or deleted by the post author, when the post is deleted, the entire comments associated to it will be deleted as well. (5 pts)

6. Hints and useful resources

- Bootstrap <http://getbootstrap.com/getting-started/>
- For your reference, the PHP-MySQL example and function documentation can be found at these websites: <http://dev.mysql.com/doc/apis-php/en/index.html>

7. What and how to submit

- (1) All source files including your PHP handling display, database connection, SQL queries, and any HTML files for user input.
- (2) Download [putty for windowsx64](#)
- (3) Transfer files from Windows to Linux.

```
input pscp [win_file_dir] [s+[學號]]@140.123.102.98:]
```



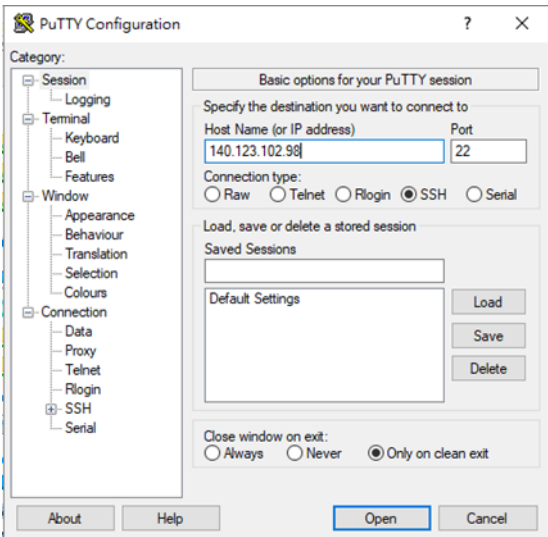
```
CA: 系統管理員: 命令提示字元
Microsoft Windows [版本 10.0.16299.192]
(c) 2017 Microsoft Corporation. 著作權所有，並保留一切權利。
C:\WINDOWS\system32>pscp C:\Users\USER\Desktop\w3_hw2_606412345.rar s606412345@140.123.102.98
```

account: s+[學號]

password: s+[學號]

Your account is prefixed "s" to your student id number (e.g., s602410000). The default password is your account (e.g., s602410000). You must change your password as soon as you first login to the machine.

(4) Start putty.exe and logged in, and the default path is C:\Program Files\PuTTY



use the s+[學號] as the account to be logged in.

(5) Input the following instruction to complete the upload task.
turnin -v -c w3ta_2018 -p hw2 [filename]

※ Grading (for TA)

Graders will test your homework only on Google Chrome.

The TA(s) will mark and give points according to the following table:

MySQL	Student should have at least two SQL tables that are connected.	10%
Main website	A website that allows posting and commenting for registered users. The home page must have rotary weather information and check weather buttons. Posting include texts and an image.	85%

	The page should have at least a sign-in page, a page for errors, a registration, and a post page, as well as a home page. All pages should serve their purpose.	
Additional Features	More features should be implemented. The more and the more sophisticated the features are, the more points for the student up to 15%.	15%
README	<p>Please provide a README files that briefly describes the details of the organization of your program and how it works. The Readme should describe:</p> <ul style="list-style-type: none"> (a) The details of the organization of your program and how it work. (b) The design of your database schema (c) The details of the advanced features and instructions to use your advanced features; <p>Otherwise, if any problem occurs, we will not accept any advanced features.</p>	5%