Web Design (COMP 20030)
Practical 6

PHP

This week, we will take a closer look at PHP. PHP is a powerful tool for making dynamic and interactive Web pages. PHP is the widely-used, free, and efficient alternative to competitors such as Microsoft's C#.Net.

Unlike JavaScript, browsers don't understand PHP. Serve up PHP to a browser and it will assume it is just text. While PHP code can be mixed in with HTML, ultimately a computer (specifically a server) will have to interpret it before you have a page fit to be parsed by a browser.

This means that pages with PHP are not HTML pages, though they may well have HTML in them. They are PHP pages: they are not called mypage.html, but instead mypage.php.

Clearly we need some additional way to deal with these php pages than what we have already encountered. Specifically, we need a web server to translate these php pages into something a browser would understand.

In this practical you will learn about PHP, and how to execute scripts on your server.

Notes for this practical:

There are lots of excellent PHP references and tutorials on the web. php.net is an absolutely invaluable resource (featuring detailed community based examples of all php functions)
Netuts+ is also worth a shoutout: http://code.tutsplus.com/categories/php

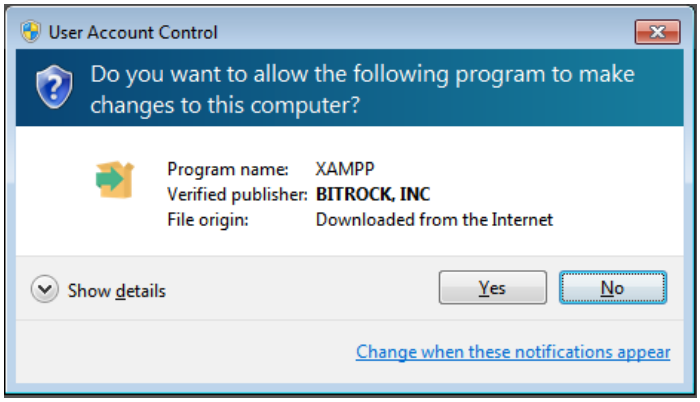
If you already have a webserver on your machine that you have been using (such as IIS) feel free to use it instead of XAMPP.

Setting up a local Server.

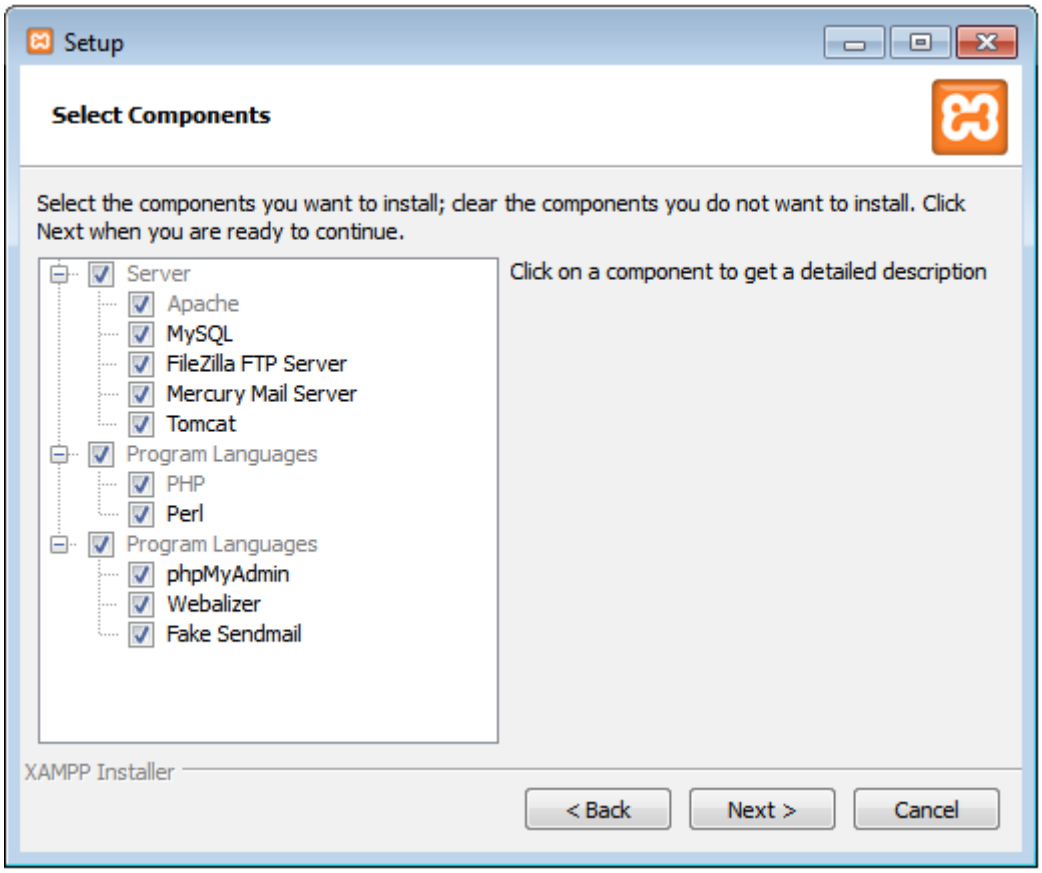
First you should download a development stack. WAMP and LAMP are popular (for Windows and Linux respectively). Let's go with XAMPP, which works across all platforms.

XAMPP can be downloaded from www.apachefriends.org/

Once it is downloaded, install the package. XAMPP will include PHP, so there is no need for separate downloads or installs.

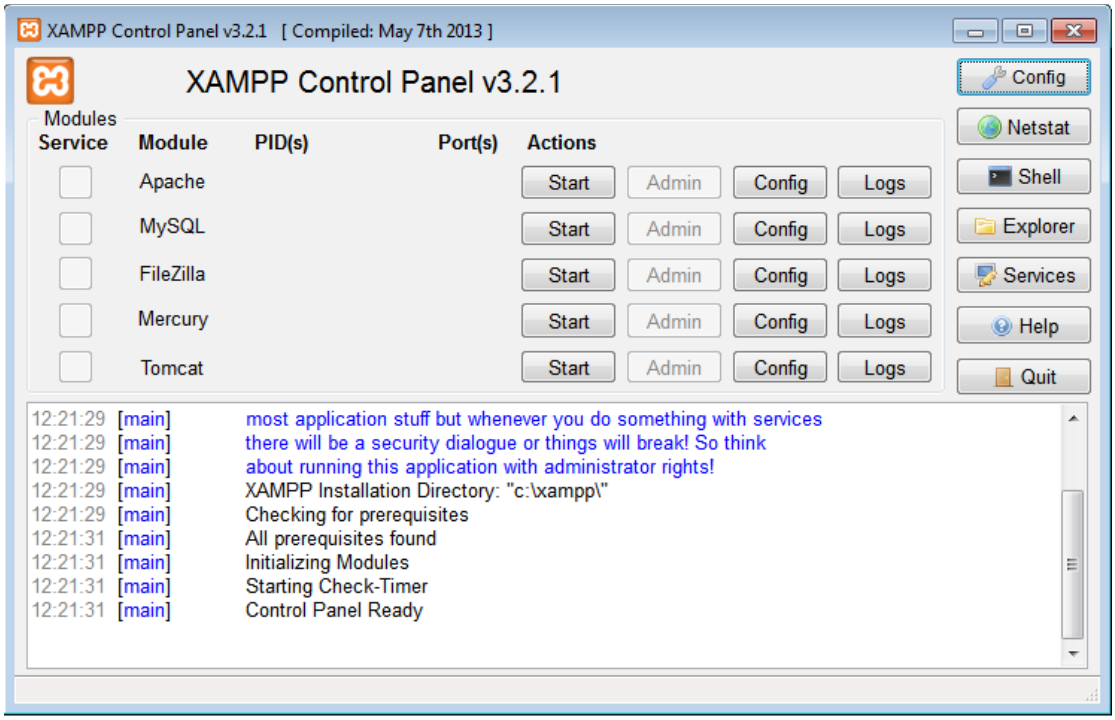


Although strictly speaking we only need Apache and PHP for this assignment, there is no harm in leaving the defaults ticked and installing all aspects of XAMPP



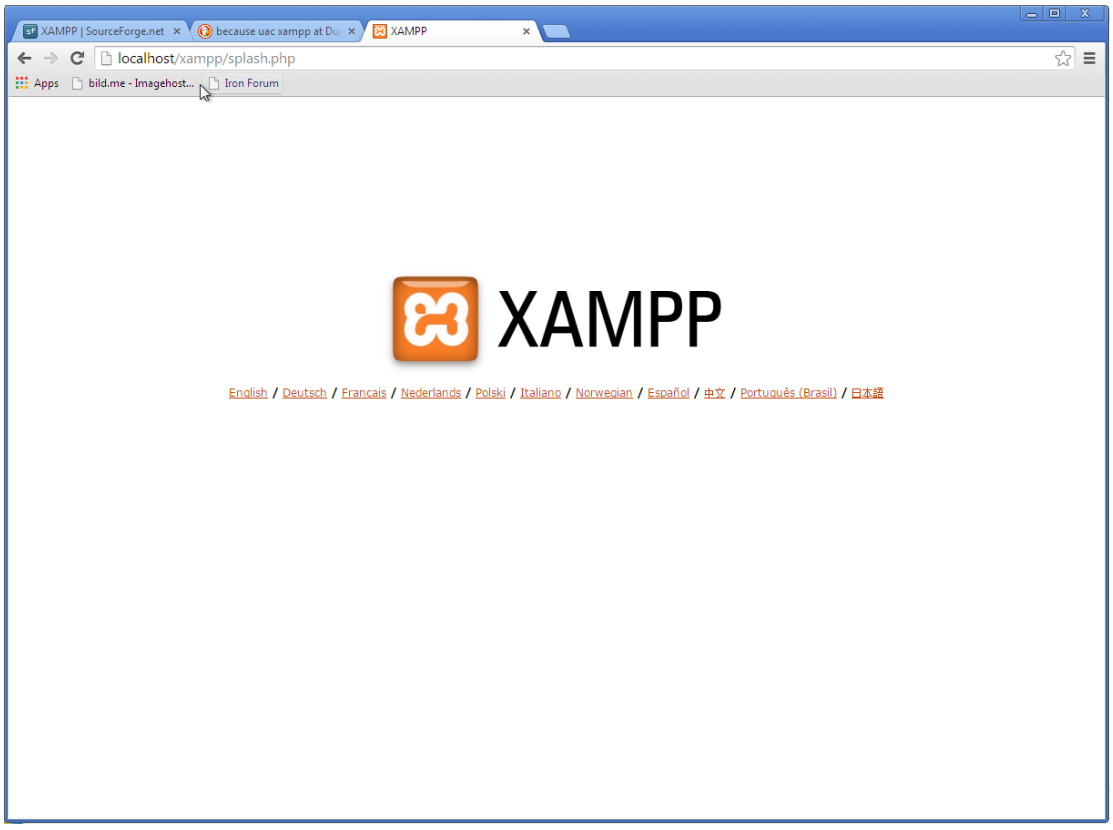
It is generally a good idea to install it directly on you harddrive (e.g. C:\), and not a sub folder (like Program Files).

The first time you run Apache you may be asked whether you want to allow your Firewall to unblock its access. You do not need to grant it access (as it's unlikely that you will ever have people visiting a website setup on your computer) but there's no harm in saying yes.



Once you have started Apache, open up a browser and type in localhost in the address bar (or 127.0.0.1)

If everything has gone correctly the XAMPP splashscreen should appear. If your browser tells you that the page cannot be found, make sure you have Apache started.



Practical Exercises

Exercise 1

Setting up a local server.

Download and install XAMPP

Find the **htdocs** folder in your local server on your hard drive. In it create a subfolder called “Practical7”.

Make a page called test.php. In it write:

```
<?php
phpinfo();
?>
```

Place test.php in the Practical7 folder.

Using your Apache Web Server, view the rendered page in your browser.

Screen-shot the result and save as q1.**png** or q1.**jpg** and place it in your Practical7 folder.

Exercise 2

In your Practical7 folder in htdocs:

Create a page called header.php which only contains the following heading:

```
<header id= “standard-head”>
<h1> My PHP pages</h1>
</header>
```

Create a footer file called footer.php which only contains today’s date
(Hint**:** try the PHP date() function).

Now make a php page, called **q2.php** that has a regular HTML5 template (like doctype, head, etc.; you can copy paste the one given way back in assignment 1). In the body create a html paragraph containing the text “Regular Website Body Content”.

In **q2.php** use the PHP function include() to import header.php and footer.php. The header should go above the words “Regular Website Body Content” while the footer should go below it.

Update these three files in your Practical7 folder within your htdocs directory on your local server, and make sure the page q2.php is displayed correctly in your browser.

Exercise 3

Using the standard HTML5 template found in practical 1, create a php page called q3.php

In the php code create a variable called $fname what will contain your first name. Create a second variable called $sname that contains the string of your second name.

Create a third variable called $fullname that will be the concatenation of $fname and $sname. Echo or print $fullname.

Using PHP conditional statements:

- If the first letter of $fname is the same* as the first letter of $sname display a paragraph that says "That name is alliterative."

  *This does not need to be case insensitive.

- If the string length of $fullname is shorter than 7 characters, display a paragraph that says "That's a short name"

- Else if the string length of $fullname is equal exactly to 8, display a paragraph that says "That's a moderately long name".

- Else display a bolded paragraph that says "That's a long name!"

Finally, edit the code of the final else condition so that "That's a long name!" is output three times, using a loop. Edit $fullname, as necessary, to test that the code works correctly.

Exercise 4

Using q4.html from practical 1 (copy paste q4.html into your htdocs folder)…

Give the form the action of "q4.php" and the method of "GET".
Make a new page called q4.php.

In q4.php, using a while loop and the each() method, echo everything that is entered into the form .
(Hint: Check the lecture notes. Also, ensure that each form element has a name)

Ensure that each element of the form displays on a new line (by using either paragraph or break line tags)

When the user clicks the submit button in q4.html they should be sent to q4.php and see output that looks like:

firstname: Joe
secondname: Bloggs

Submission Instructions

Create a zip file containing all of this week's solutions.

In Windows you can select the "Practical 7" folder that is within htdocs, right click and choose send to > Compressed (zipped) archive.

In Mac OS X, you can select the "Practical 7" folder that is within htdocs, right click and choose Archive.

Name your zip file in the format of "*name* – Practical7 – *studentnumber.zip*", replacing name and *studentnumber* with your own name and student number. e.g. "Bloggs, Joe – Practical7 – 12345678". Upload your zip file using the Practical 7 submission form available on Moodle.