Web Design (COMP 20030)
Practical 5

JavaScript II & Web Servers

This week, we will start to take our first look at web servers while also testing additional JavaScript functionalities

Notes for this practical:

As usual, your HTML should validate, and your scripts should run without errors.

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.

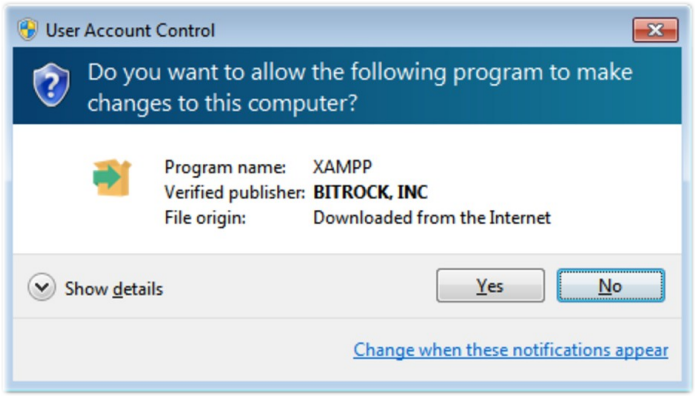
Your JavaScript can be added as an external script or written within <script></script> tags in your page for this assignment, but if you are importing an external JS file you will find the "defer" attribute useful.

Setting up a local Server.

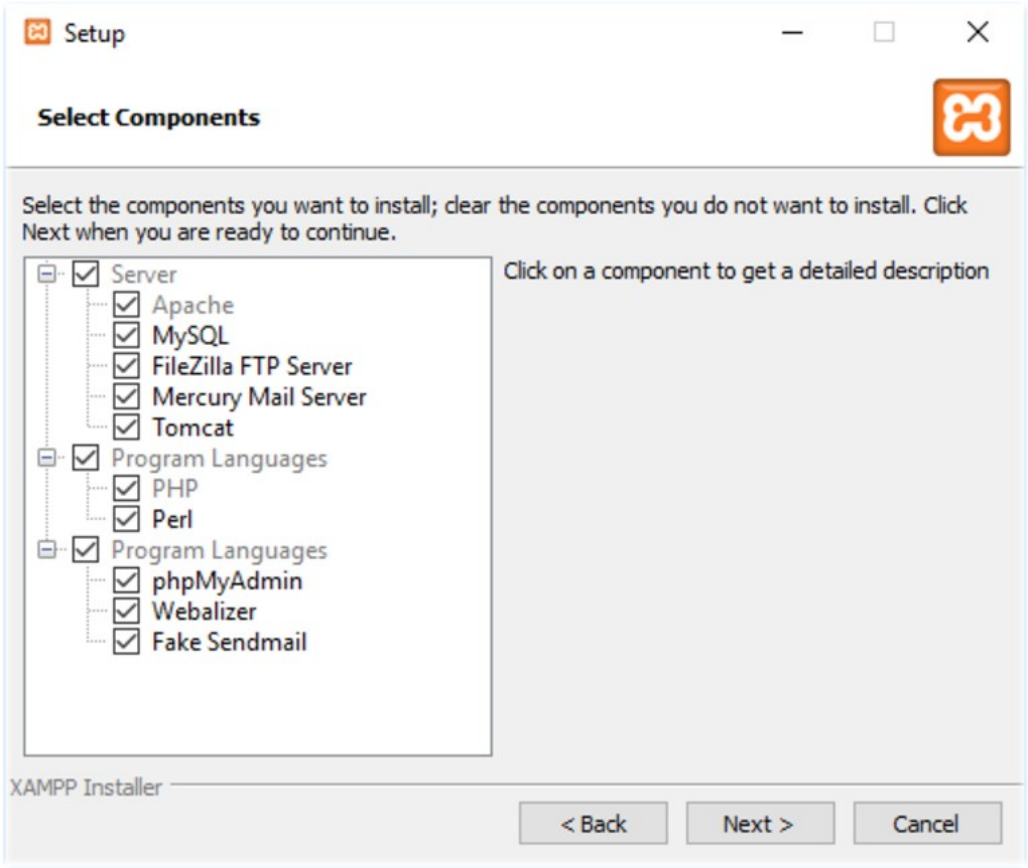
First you should download a development stack. WAMP and LAMP are popular (for Windows and Linux respectively). Let's go with XAMPP, which works across all platforms.

XAMPP can be downloaded from https://www.apachefriends.org/index.html

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

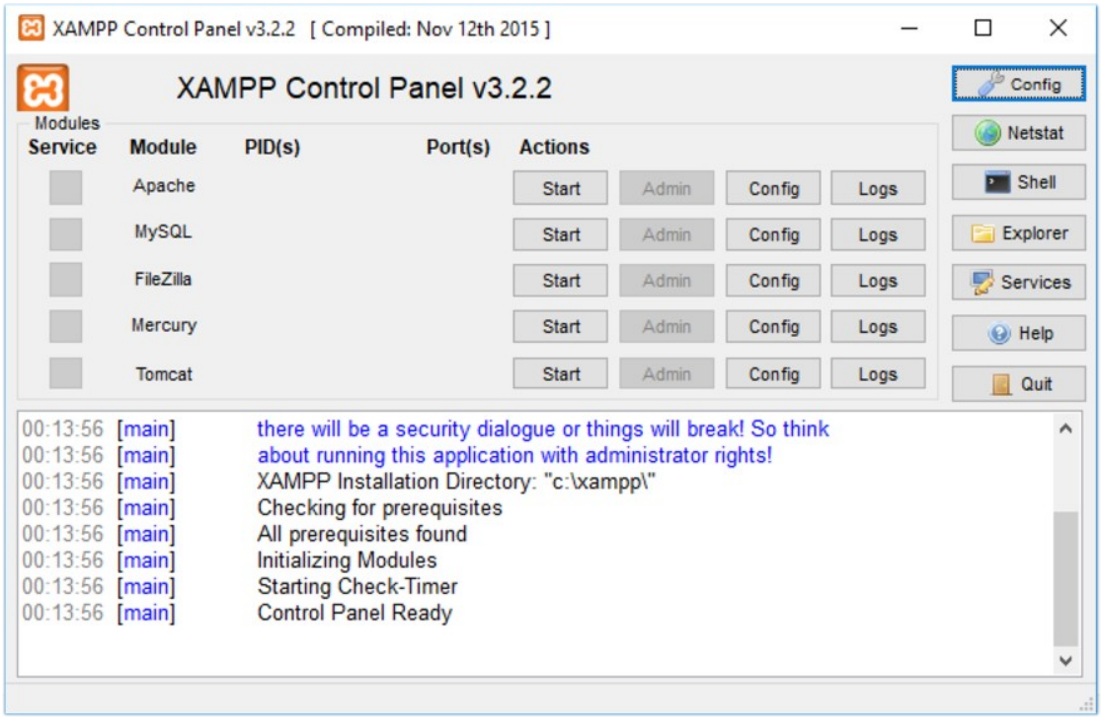


There is no harm in leaving the defaults ticked and installing all aspects of XAMPP



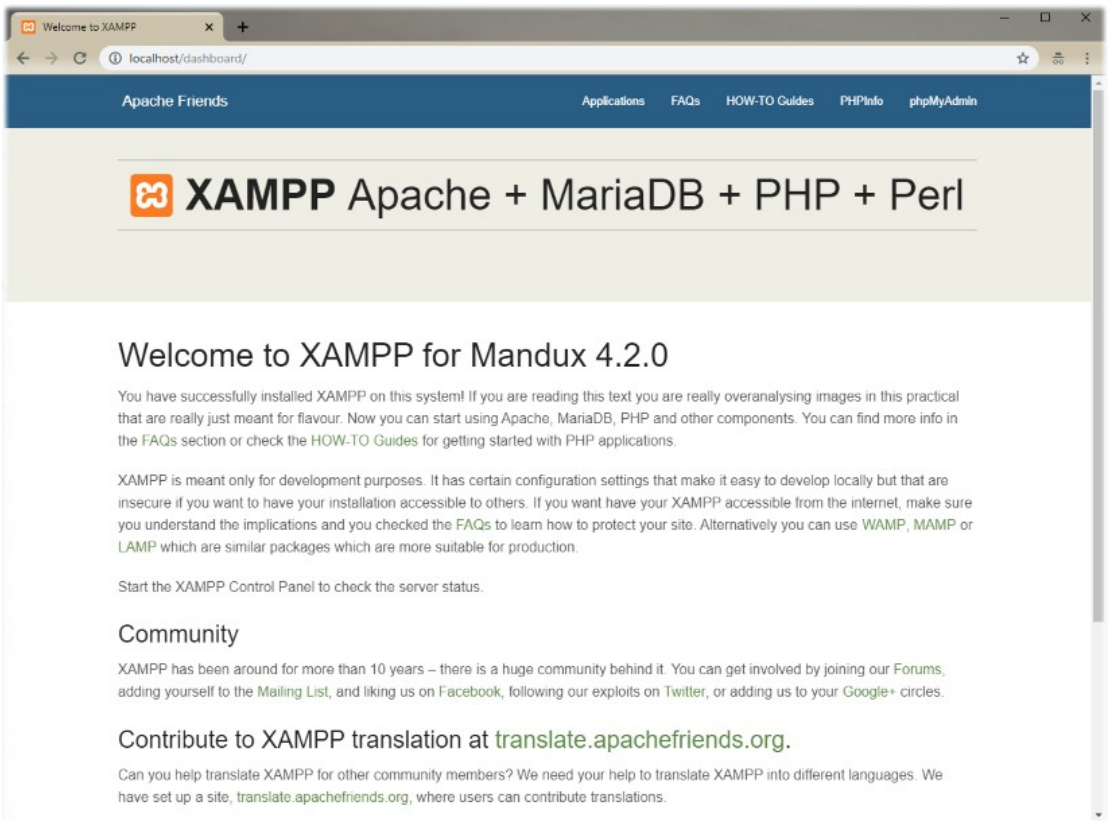
It is generally a good idea to install it directly on you harddrive (e.g. C:\), and not a sub folder (like Program Files).

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. Note that the installation choices and control panel on Mac may have fewer options – this is fine (but the UI may look slightly different).



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 “Practical5”.

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

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

Place test.php in the Practical5 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 Practical5 folder.

Exercise 2

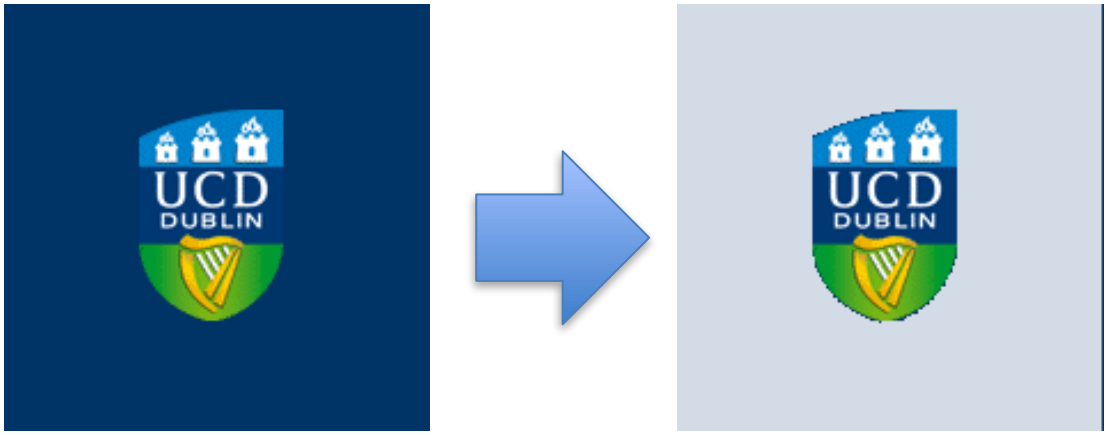
Make a webpage that contains a canvas element 300*300 with the id of “thisCanvas”. Add a button anywhere on the page. When the button is clicked a function will cause a line to appear in the canvas, going from the bottom left corner to the top right. You’ll have to make this function.

Add a second button. Similarly, when this button is clicked a function should draw a rectangle on the canvas. The position of this rectangle can be defined using canvas.width and canvas.height.

Add a third button. This should clear anything that is currently on the canvas using the clearRect() function (this is a library function, you don’t have to make it).

Save as q2.html

Exercise 3



1. In this exercise you will create a rollover using the UCD crest as an image. Download the two sample images to a Practical5/images folder. The sample images are to be found at on BrightSpace and are called image1.gif and image2.gif.
2. Prefetch the two images using link rel="preload"
3. Write the body of the web page to contain a single image – the preloaded image1.gif. Make the image link to the UCD home page.
4. Add an event listener for a mouseenter based event that swaps image1 for image2. On mouseout image1 should show again. This can be achieved by changing the value of the img source

Save as q3.html

Exercise 4

The next question is a little different. In this question you must access a JSON file (colorData.json available on BrightSpace) and output its contents into a HTML table.

For this to work the following code will be useful. Note that for this code to work you must access it through the **web server on XAMPP** (otherwise you are likely to get a failed Access to XMLHttpRequest error due to your browser's CORS policy).

You will need to write some JavaScript to loop through the parsed JSON contents and output this in the form of HTML. You find the code written below here, for easy copy-pasting.

```
// 1. Create a new XMLHttpRequest object
let xhr = new XMLHttpRequest();
```

```
// 2. GET the JSON file (must be specified)
xhr.open('GET', THE-JSON-FILE);

// 3. Send the request over the network
xhr.send();

// 4. This will be called after the response is received
xhr.onload = function() {
  if (xhr.status != 200) { // analyze HTTP status of the re-
sponse
    alert(`Error ${xhr.status}: ${xhr.statusText}`); // e.g. 404:
Not Found
  } else { // show the result
     let parsedObj = JSON.parse(xhr.responseText);
        // This function is defined below and deals with the JS
ON data parsed from the file.
     put_your_code_in_this_function(parsedObj);
  }
};

function put_your_code_in_this_function(parsedObj){
  // parsedObj.name-within-JSON-for-values-you-want-to-access
  console.log("This doesn't do anything yet");

}
```

Save as q4.html

Submission Instructions

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

Submit on BrightSpace