Lab 01: HTML Exercises 1

Mercury, Introduction to HTML,

Tutorials:

Completing the lab exercises will help you gain the skills and knowledge you need for this unit and its assignments. The exercises will be available on Canvas well in advance of their due date.

Lab exercises will be checked by your tutor before they are due.
You are encouraged to complete the lab work prior to your tutorial so you have a chance to discuss it with your tutor and fix any problems before it is assessed.

Lab work will only be assessed if it is run from the Mercury server, and in the specified directory (e.g. lab02, lab03, etc.).

Aims for this Lab:

- To set up your user account on the university's Apache webserver
- To create HTML5 pages using a text editor
- To develop an understanding of the client-server environment and the processes we will be using throughout the semester to load pages to the Apache webserver, and to view them through a browser.
- To validate HTML pages – locally on your computer, and on a server
- Gain some skills and knowledge to help you develop Assignment 1.

Task 0: Set up and connect to Swinburne VPN

If you are doing the labs off campus, you will need to connect to Swinburne VPN to access the Mercury server. Please refer to https://www.swinburne.edu.au/life-at-swinburne/student-support-services/study-learning-support/student-systems-hardware-software/ for instructions.

Task 1: Set up and test your user account on Mercury

Step 1: Log onto the web server *and set up your account.*

Read the Guides: *"Quick Start Guide"*
linked from the Canvas > Modules > Server Access Guides.

A web account is needed to load, test and run your web pages (HTML, CSS, JS) and server-side scripts (PHP). For this purpose, every student is assigned a Web server account on the Apache server at ***'mercury.swin.edu.au'***.

Using 'PuTTY', a SSH (Secure Shell) client, (available on most lab computers under Start>Programs>Network), log in to the mercury web server with username, s<your 7-digit or 10-digit Swinburne ID>, (for example, s1234567) and your SIMS password.

The first time you log in, the required directories will be created and the URLs will be shown.
Once the directories are created you can upload files using **WinSCP**, **FileZilla** or any other **sftp** client.

Step 2: Connect to the web server to upload files.

Read the Guide "Transferring files to/from Mercury"
linked from the Canvas > Modules > Server Access Guides.

To test our Web pages, we need to first place them on the mercury server. We can use '**WinSCP'** a Secure Copy and Paste file transfer client (available on all lab computers under *Start>Programs>Network).*

Using WinSCP, log in to the web server, mercury, with the host name '*mercury.swin.edu.au*',
user name ***s****<your 7-digit Swinburne ID>* and SIMS password.

After a successful log in, you will be able to see the unit folder. It will be named according to your unit **cos10011**. Note that the "cos" is in lower case. Unix filenames are **case sensitive**.

If you are using a Mac there are some instructions for using Cyberduck under the Server Access Guides menu on Canvas.

Step 3: Create a folder (directory) to contain your web pages.

Create a folder for each web project. Create a folder named 'lab01' to contain all the files that you develop in this lab.

Note: Apache web servers, such as 'mercury', are case sensitive. It is recommended that you use **lower case** and avoid non-alphanumeric characters such as a space when naming folders.

Using WinSCP (or the file transfer client of your choice), create a folder '**lab01'** under ~/*<your_unit_code>/www/htdocs* folder in mercury.

Step 4: Create a web page for testing.

Use a code text editor on your local computer (e.g. NotePad++) to create a file named **myhtml.html**

HINT: Code editors like NotePad++ highlight the different parts of your text according to what language is being used. For example, in an HTML file NotePad++ highlights **element tags blue**, **attribute names red**, **comments green**, **content text black**, and so on. The editor knows what language the file is being written in by looking at the extension – e.g. .htm, .html, .js, .c , .php, etc. These colours not only make the text easier to read, but can help you pick up keyword spelling errors and other syntax errors.

So save the file with its extension before you start typing.

Type the following code into your newly saved file, and then resave it:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <meta name="description" content="Creating Web Applications" />
  <meta name="keywords"    content="HTML, CSS, JavaScript" />
  <meta name="author"      content="put your name here" />
  <title>My First HTML 5 webpage</title>
</head>
<body>
  <!-- First HTML5 Example -->
  <h1>Creating Web Applications - Lab 1</h1>
  <p>Can I use .htm instead of .html as my file extension?<br />
  <em> Yes.</em></p>
  <p>Are folder and file names case sensitive on Apache servers?<br />
  <em> Yes.</em> <strong>Best to use only lower case.</strong></p>
</body>
</html>
```

Yes! Type your name here, you are the author of this webpage

Step 5: Transfer your web page to the web server.

Read the Guide "Transferring files to/from Mercury"
linked from the Canvas > Modules > Server Access Guides.

To be able to run your Web pages, you must place the files into ~/*<your_unit_code>/www/htdocs* folder on mercury or any subfolders you create.

Using WinSCP, drag and drop (or copy and paste) your file 'myhtml.html' from your local machine to the ***htdocs/lab01*** folder on the server.

Step 6: Test and view web pages.

Read the Guide: "Viewing files in Mercury"
linked from the Canvas > Modules > Server Access Guides.

To view the pages through http, use Firefox and type in the following address,

https://mercury.swin.edu.au/<your_unit_code>/<username>/<folder/.../filename>

The <*username*> is **s**< *your 7-digit or 10-digit Swinburne ID* >. For example

https://mercury.swin.edu.au/cos10011/s1234567/lab01/myhtml.html

When the ***authorization request dialog*** pops up, use your ***SIMS username and SIMS password*** to confirm access.

Note that the path to the 'htdocs' folder on the server is 'mapped' to a different path in the URL.

Notes:

- Step 1 is only done once.
- Steps 2 and 3 are performed every time you start a HTML/CSS/JavaScript development session.
- Steps 4 to 6 are performed repeatedly in sequence when creating, testing and debugging your HTML/CSS/JavaScript and PHP codes.

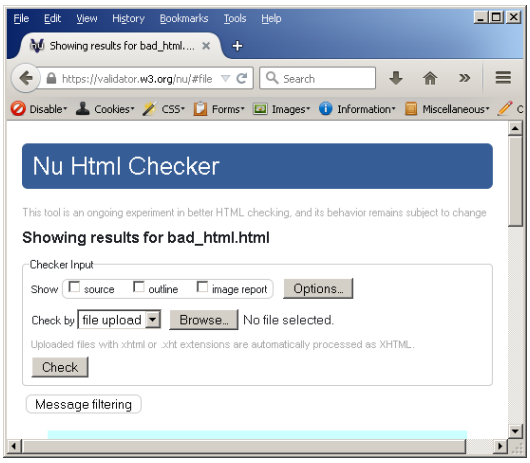
Problems:

- For major access problems, such as no account, contact the ITS Service Desk on 9214 5000 or ServiceDesk@swin.edu.au and inform them of your situation.
- For problems in completing steps 2 to 6:
  - refer to the online guides under "Server Access Guides" on Canvas
  - ask your tutor for help
  - or visit the programming help desk

Task 2: Validate your Web Page as HTML5

Open up the W3.org validator web page at https://validator.w3.org/nu/

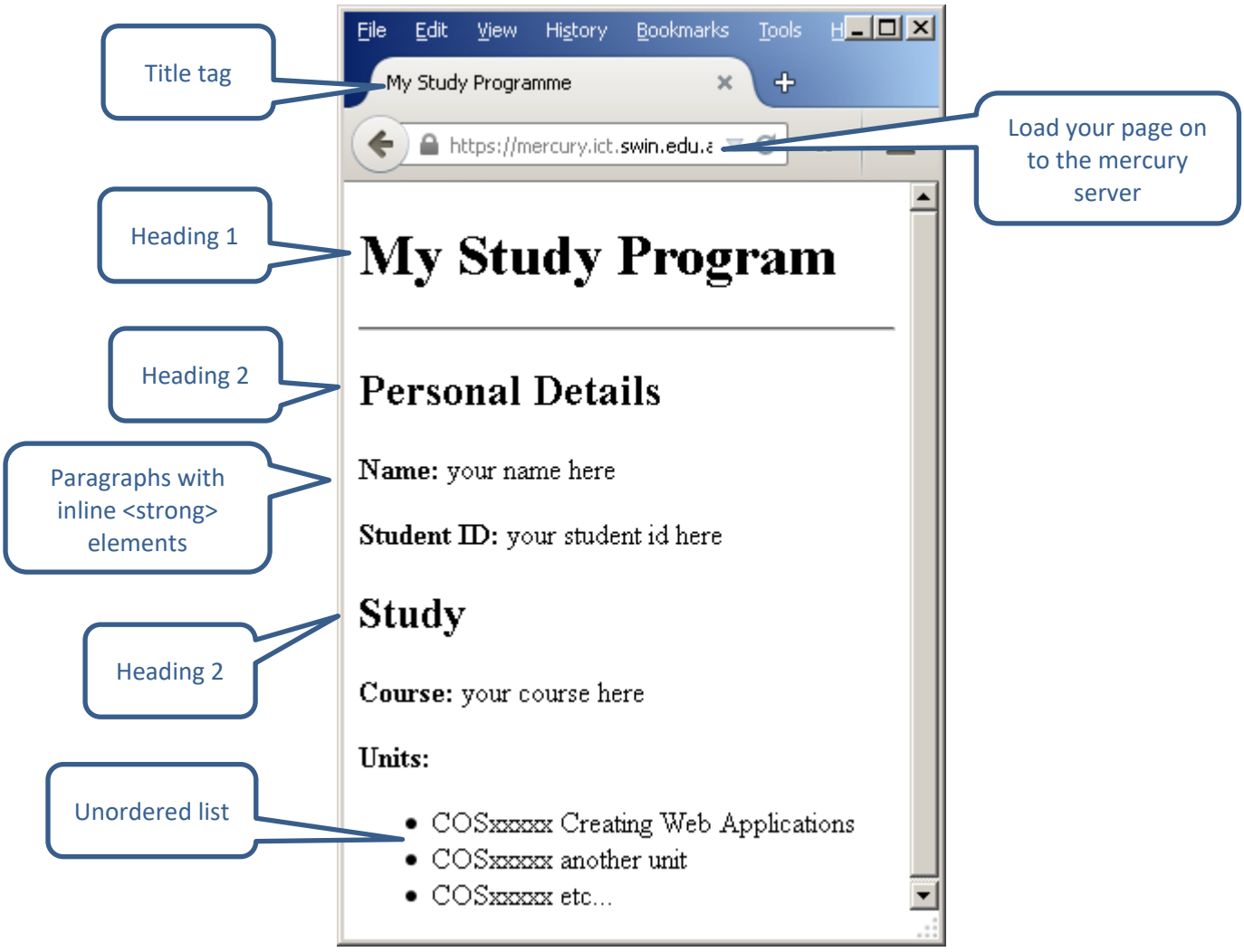
- Select the **File Upload** option from the drop down list box as shown below.
- Upload the HTML file you created in Task 1.
- Check for any validation errors and fix them.
- … repeat … until no errors.



Task 3: Create a Web page about you

Step 1: Create the HTML

Use any text editor on your local computer (e.g. NotePad++) open a new file and save it as **`mystudy.html`** in local folder **`lab01`**.



Check Task 1 Step 4 above if you have forgotten how to do these steps

Step 2: Create the HTML structure and header

- At the start of the html text file add a document type declaration
- Create a `<html>` root element with a `<head>` and `<body>` children
- In the `<head>` element create meta-tags for
  - charset
  - description
  - keywords
  - author
- Give the page a title using the appropriate element

Step 3: Create HTML content

- Within the <body > element, create the Web page as shown above with your name, student id, course and units set out as shown. You may need to consult the lecture notes for some of the elements.

Step 4: Validate your html source file using the W3C validator (as in Task 2)

Step 5: Copy the html source file to Mercury server

- Load the page onto Mercury
- To view the pages through http, use any Web browser and type in the following address,

```
https://mercury.swin.edu.au/<your unit code>/<username>/<folder>/<filename>
```

The <*username*> is **s**< *your 7 or 10 digit Swinburne ID* > and <your unit code>. For example

```
https://mercury.swin.edu.au/cos10011/s1234567/lab01/mystudy.html
```

When the authorization request dialog pops up, use your ***SIMS username and password*** to confirm access.

[IMPORTANT] Send your tutor the link to your web page running on the Mercury server to be marked off.

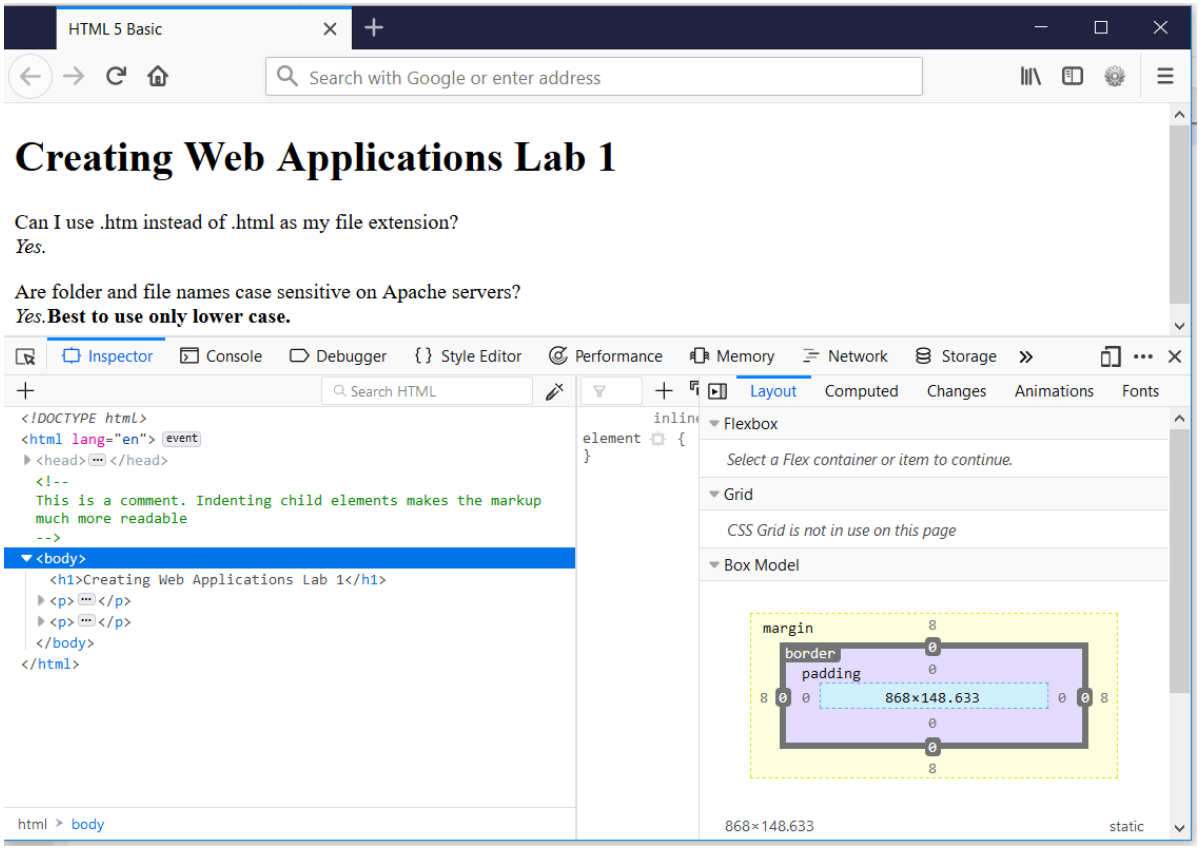
Task 4: Firefox Web Developer Tools

Built-in Tools

Firefox has a wide range of built-in tools that can assist Web developers.
See https://developer.mozilla.org/en-US/docs/Tools

Load the Web page **`myhtml.html`** you created in Task 1 into the browser, right click on the web page, and from the context menu click **`View Page Source`**. A window will appear showing the source code you have written.

Load the Web page **`myhtml.html`** you created in Task 1 into the browser. Click the Firefox menu (Three horizontal parallel lines) at the top-right of your browser window. Select Web Developer > Inspector. You will be able to see a structured tree view of your html file as shown below.



Add-ons (optional)

Firefox is open source software, and many useful 'Add-ons'/ 'Extensions' can be installed into the basic browser. In particular, the **'Web Developer Extension'** tools are probably the most useful, particularly to enable you to check the quality of your HTML and CSS markup.

To **Install** the Add-on / Extension:

- Start Firefox
- Click the Firefox menu (Three horizontal parallel lines) at the top-right of your browser window. Select "Add-ons".
- Click the 'Find more Add-ons' link.
- Search for an extension (for example, 'Html Validator') and install it.
- Explore how to use this extension.