Student L8A4

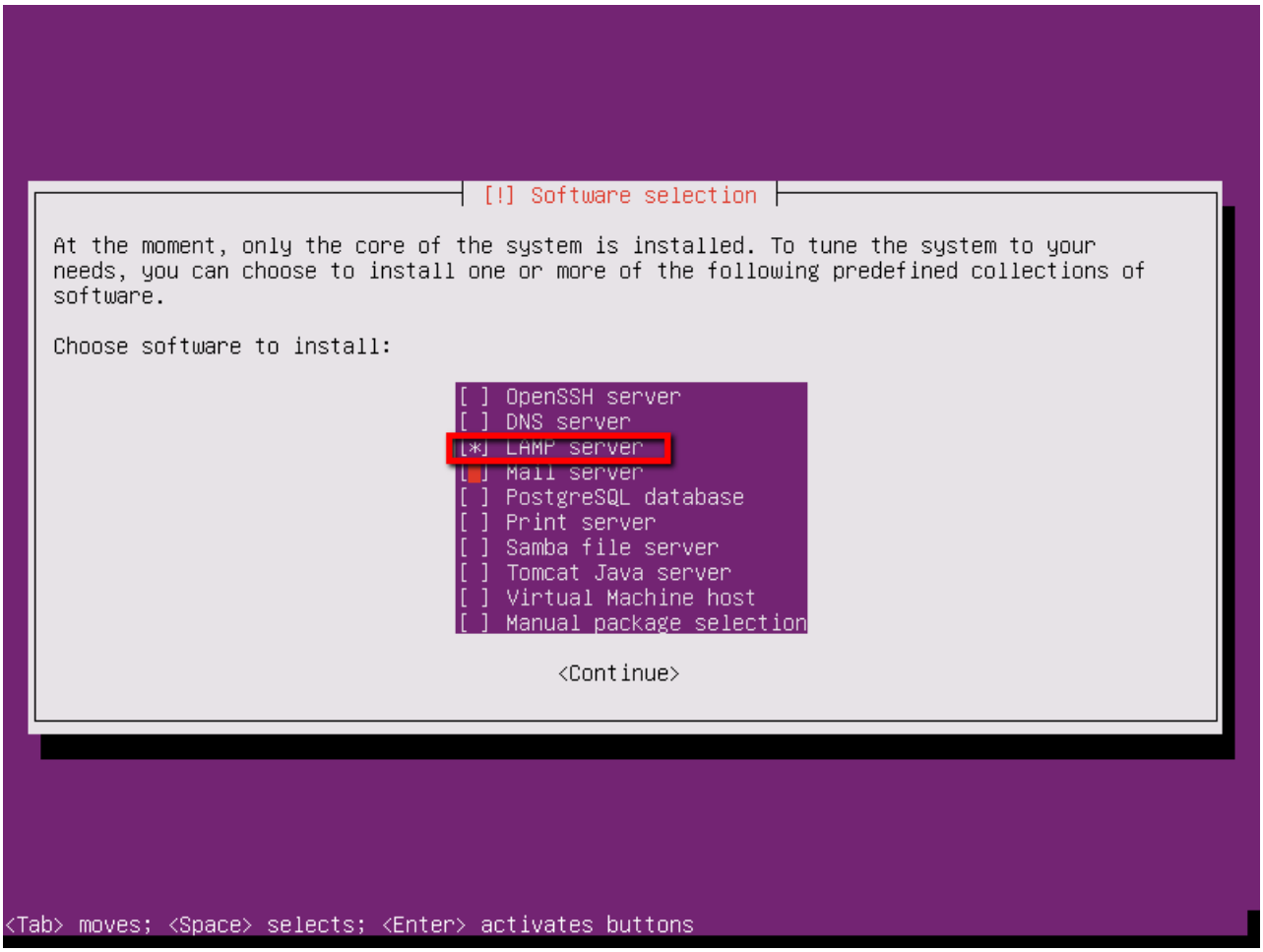
In this activity, you will be setting up a web server and creating a couple pages to gain familiarity with a few different Hyper Text Markup Language (HTML) tags.

Creating a Web Server:

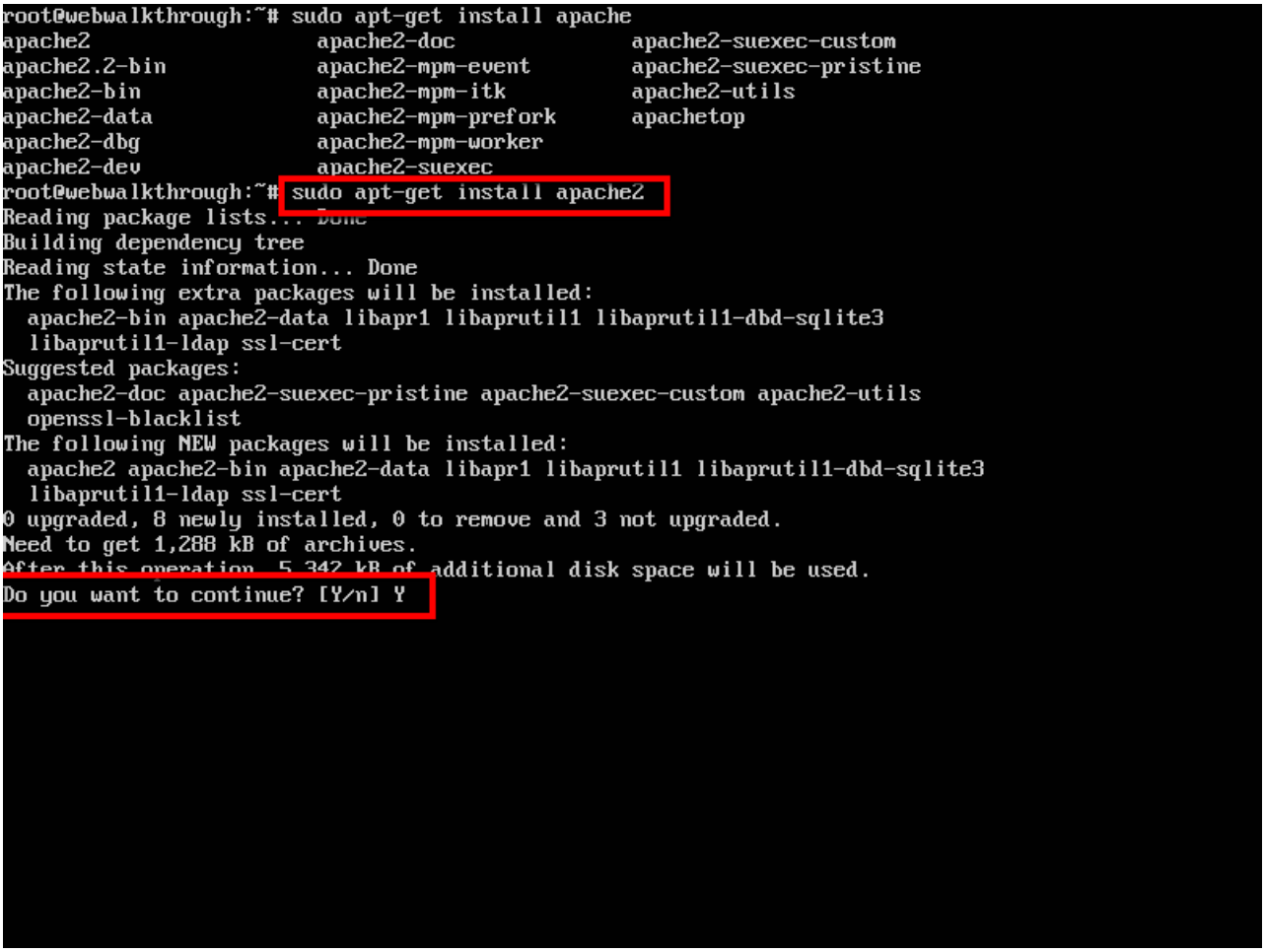
This tutorial details installing Apache on an Ubuntu Server distribution (not an Ubuntu Desktop distribution) via the command line/terminal. Other Linux distributions will follow similar instructions with slight variances.

If you are starting from the installation of the server start at step 1 and then skip to step 3, otherwise if the server is already set up, start at step 2.

1. At the Software Selection prompt, select LAMP server from the list, along with any other software you may want.

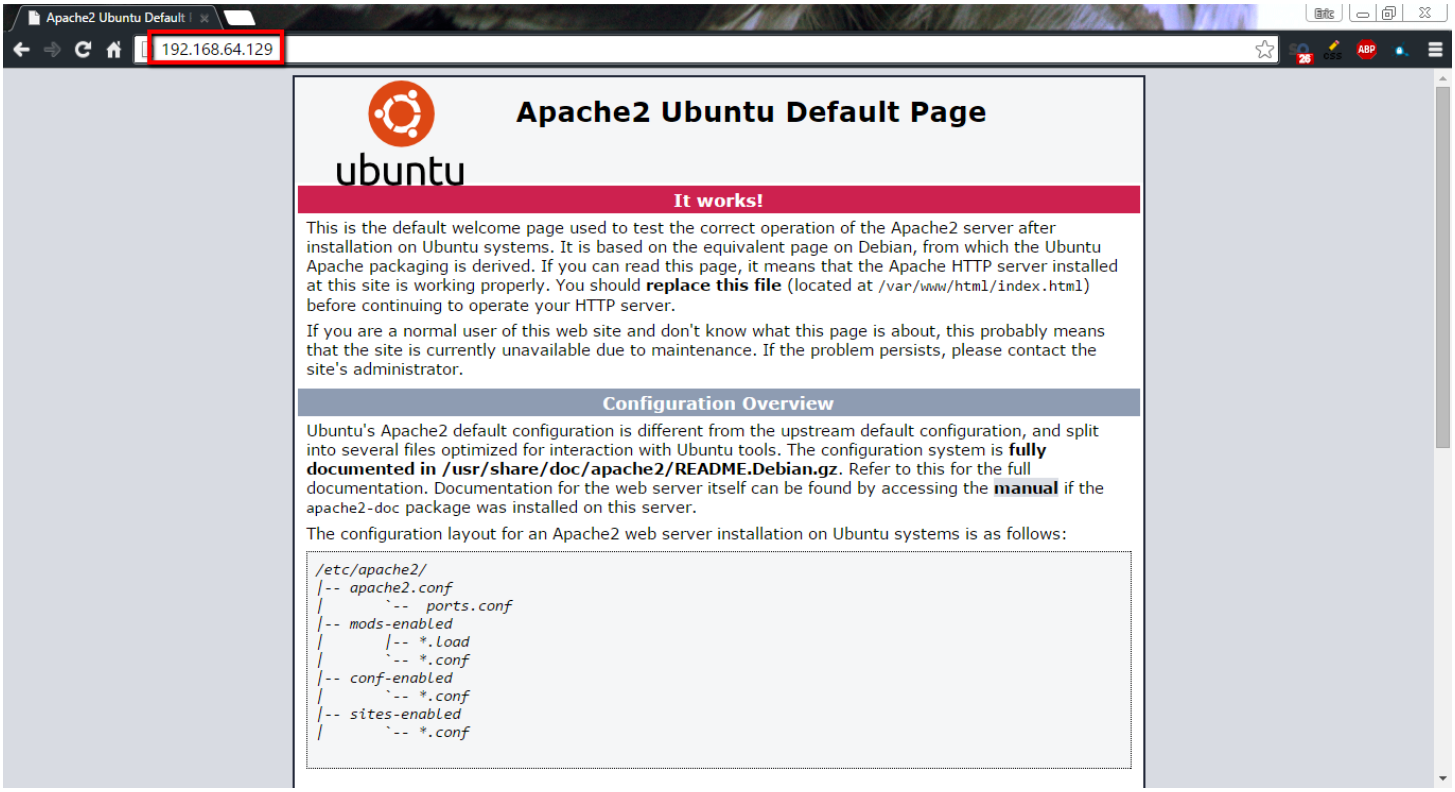


2. If you are starting from a running Ubuntu server, then using an account with Sudo privileges, run the command `sudo apt-get install apache2` . When prompted, type the letter “Y” to accept and install Apache.

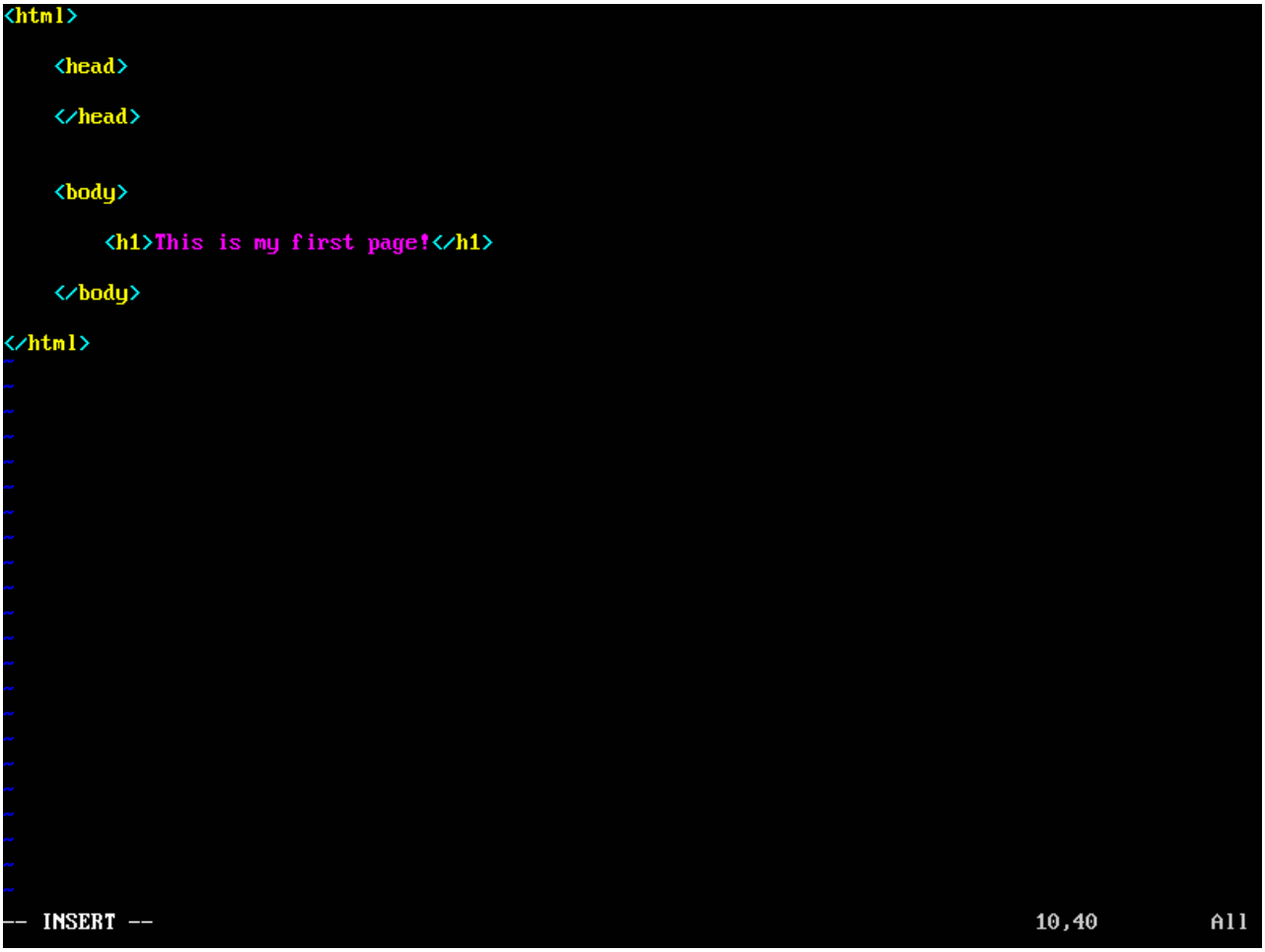


3. After the installation is complete, use the command `cd /var/www/html/` to change to the web server directory. If you run the command `ls` you should see a file named `index.html` . Any time you go to an IP Address or Web Address, the server looks up a file named index.* (where * can be html, php, jsp) and that is the page that is displayed. Unless you specify a certain page.

4. If you use a browser to go to the IP Address of your server, you should see the default Apache web page as shown below. (You can find the IP Address by using the command “ifconfig”)



5. Open the `index.html` file in your editor of choice (nano, vim, emacs, etc.), clear the file and replace it with the code from the screenshot below. It is necessary for most html files to have an <html> tag, <head> tag, and a <body> tag. It is also imperative for most tags to be accompanied by a closing tag which takes the form of </"tag_name"> as you can see in the code. In this code, the <h1> tag creates text with the styling of a level 1 header. It will display the text between it's tags.

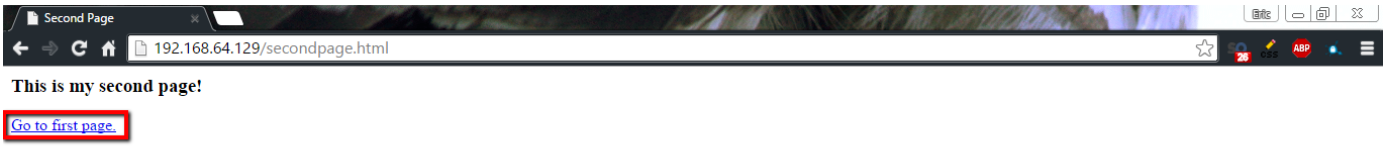


6. Save the file and return to your web browser and refresh the page. You should see the new webpage that you just created.
7. Next we are going to create a second page with a couple different tags. Open a blank file in your editor of choice and name it `secondpage.html` . Add the code in the screenshot below to the new file. The <h3> tag is similar to the <h1> tag from the last file, but it gives the text the style of a level 3 heading which is smaller than the level 1 heading. Another new tag is the <title>. This tag sets the title of the page as it appears in the Tab or at the top of the window. The third new tag is the <a> tag, which stands for anchor. It creates a hyperlink to another location. This one is directing us back to the first page. The “href” attribute sets the destination of the link.

```
<html>
    <head>
        <title>Second Page</title>
    </head>

    <body>
        <h3>This is my second page!</h3>
        <a href="index.html">Go to first page.</a>
    </body>
</html>
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                                    14,51         All
```

8. Save the file and go to the address “http://your_ip_address/secondpage.html”. It will direct you to the page you just created. Pay attention to the name at the top of the window or tab it should say “Second Page” or whatever you put in the <title> tag. If you click on the link, it will take you back to the first page.



Congratulations, you have successfully set up a web server and created a couple pages. There are many different tags that you can use when creating web pages. A good resource to find more and find out how to use them is: http://www.w3schools.com

If you have your DNS server working, and you have set up the DNS entry for your web server, usually a www entry, you can go to `www.#your_fqdn` in your web browser and you should be able to see your web site. If you do not have DNS set up, you can go to your server's IP Address by typing `http://#your_IP_Address`
