UCSCx
EMBD-X413
Fall 2017

Assignment #4 – Dynamic Web Page For Webserver (20pts)

Overview:
For this assignment, you will be using a programming language (Python) to dynamically create a web page that displays your Pi's sensor data whenever it is executed. Like assignment #1 & #3, your output screen will display basic weather-related information in a web page. This time, you will have to use your assignment #3's HTML as a template. You will have to also include a current date-time stamp whenever your Python code recreates your web page. For time sake, you can run your program from the command line to dynamically create your web page.

Requirements:
1. Assignment #3's HTML code.
2. Use a Python program to dynamically create your assignment #4's sensor info.
   2.1. Save Python file as: assignment4-<your initials>.py. Example: assignment4-gg.py
3. Web page must be dynamically created after running your python program as a HTML file.
   3.1. Your Python code must create (and replace) your assignment4.html page content in your Pi's /var/www/html/ directory.
4. Web page must have a current date time stamp that is refreshed when the page is dynamically recreated.
5. For the date time stamp, you can use Python's built-in now() function.

   ```
   import datetime
   dt = datetime.datetime.now()
   write ("Current date &amp;amp; time = " + dt.strftime('%Y/%m/%d %H:%M:%S') )
   ```
6. Web page must contain at least 3 sensor outputs.
   6.1. Temperature, Humidity, and Pressure
7. Sensor data must be tabulated for readability.
   7.1. Use a side or top header column to explain your sensor data.
8. Web page must be readable using a standard popular web browser. Please double check your work BEFORE turning it in! This includes extra credit design elements.

Procedure:
First, write down your Pi's IP address. Use a text editor or a Python IDE to create your Python program. After completing your edits, run your program from the command line. Use a browser to view your dynamically created file in your Pi's web server.

Grading:
Grading for this assignment will be based on meeting the basic requirements. You will turn in BOTH of your Python source code file and one of your dynamically created HTML files by emailing it to me using the UCSC portal system.

Extra Credit:

For this assignment, you can use JavaScript AND/OR Cascading Style Sheets (aka CSS) to get extra credit. In addition, I will add a couple of points for creating a scheduled task in crontab to run your assignment #4 python program. I will add 1-5 points extra for your creative efforts AND for your crontab configuration. To get the extra credit points, be sure to include your crontab text file, JavaScript and CSS files. For your crontab file, use assignment4-<your initials>.cron. Turn in all external additional files along with your HTML and python program files to get your extra credit.