Week 7 Deliverables

Overview: This week, you studied additional Flask functionality for creating a secure login form and associated files for a web site. The Lab for this week demonstrates your knowledge of this additional Python functionality. Be sure to use the examples in the textbook reading along with the associate libraries, functions and processes when completing the assignments for this week.

Submission requirements for this project includes multiple files. (Zipping them into one file is acceptable and encouraged):

- Python Web Page Code (Python code, Templates, CSS and other associated files)
- Word or PDF file containing your test and pylint results

Python Applications for this lab: (total 100 points):

1. **(80 points)** This exercise uses your programming environment to enhance the Web site you created last week with additional functionality to include images, tables and a Form using Python flask. Specifically, you will add two (2) additional routes allowing a user to register and login to a web site. Additional security considerations include other routes (beyond the register route) will not be accessible until a successful login has occurred.

In addition to the requirements list above the following functionality should be found within your **web site** on one or more web pages.

- Add at least 4 different images. The images should be local in your environment. For example, they should be saved in your environment and referenced similar to this syntax: <img src="image.gif">
- A Table with at least 4 rows and 3 columns.
- A user registration form
- A user login form
- A password complexity should be enforced to include at least 12 characters in length, and include at least 1 uppercase character, 1 lowercase character, 1 number and 1 special character.

The content and topic of the new images, and tables are up to you. How much is required for the user registration is up to you as well. However, the registration and associated login should contain at least a login name and password.

Hints:

1. Start early. This will take you longer than you think.
2. Test all aspects of the forms from input to output on your environment.
3. Use comments to document your code
4. Test with many combinations.
5. Use pylint to verify the code style – the goal is a 10!

2. **(20 points)** Document your testing results using your programming environment. You should also include and discuss your pylint results for the application. The test document should include a test table

that includes the input values, the expected results and the actual results. A screen capture should be included that shows the actual test results of running each test case found in the test table. Be sure to include multiple test cases to provide full coverage for all code and for each function you develop and test.

Any submissions that do not represent work originating from the student will be submitted to the Dean's office and evaluated for possible academic integrity violations and sanctions.