CSCI 4000 Assignment 4

Q1: **Registration** [50 points] - PHP
Create a registration form, similar to what you might encounter when registering for an online Web site, including two sections: Personal Information, and Security Information.
In the Personal Information section, add name and e-mail address. Include default text in the name and e-mail text boxes.
In the Security Information section, add password and password confirmation fields.
Use html table (or css) to format the input fields (into rows and columns).
Add Submit button. Save the document as `<my name>Registration.php`.

Submit the form to the `<my name>RegnDisplay.php` file. `<my name>RegnDisplay.php` should use the $_POST array to get all the form fields/inputs (name, e-mail, password, confirmed password etc), and display them.

Use an external *.css file to format both .php files.

Note: For this question, name your files `<my name>Registration.php`. and `<my name>RegnDisplay.php`. For example, my files would be named: `LeongLeeRegistration.php` and `LeongLeeRegnDisplay.php`. Change the page titles to `<my name> Registration` and `<my name> RegnDisplay`.
Important: If you do not put `<my name>` to the above mentioned fields (page title and filename), **you will get 0 point for the question.**

Estimated time: 2 to 3 hours

Q2: **Average** [50 points] - PHP
Write a PHP web page (program) that asks user to enter five numbers (five text boxes). Include default values in these five textboxes. Use html table (or css) to format the input fields (into rows and columns).
Add Submit and Reset buttons. Save the document as `<my name>Average.php`.

Submit the form to the `<my name>ProcessAverage.php` file. `<my name>ProcessAverage.php` should use the $_GET array to get all the form fields/inputs (five numbers), and display them.
Add statements to calculate the average of these five numbers, and display the result (with 2 decimal places).
Add statements to display the current year/month/day/time.

Use an external *.css file to format both .php files.

Note: For this question, name your files `<my name>Average.php`. and `<my name>ProcessAverage.php`. For example, my files would be named: `LeongLeeAverage.php` and `LeongLeeProcessAverage.php`. Change the page titles to `<my name> Average` and `<my name> ProcessAverage`.

Important: If you do not put `<my name>` to the above mentioned fields (page title and filename), **you will get 0 point for the question.**

Estimated time: 2 to 3 hours

Submission instructions:
You need to test the above programs (questions) separately, and provide **two test cases** (if applicable) for each program (question). Do a screen capture of the input and related output for each test case. Use any graphic editing software (e.g. Microsoft Paint, Adobe Fireworks, GIMP) to cut out the program input and output (from the screen capture), paste them into a word document under a related question number, save the document as a pdf file.

You need to submit the following:
1. A pdf file containing the screen captures of program input and output of all test cases, name the file **lastname_firstname_assignment04.pdf**.
2. All php files. Zip your files into a single zip file (or rar file) **lastname_firstname_assignment04.zip**.

Please submit electronic copy (the above mentioned **two files**) to D2L digital dropbox.

Grading guidelines (programming questions):
Your programs will be judged on several criteria, which are shown below.

- Correctness (50%): Does the program compile (run) correctly? Does the program do what it's supposed to do?
- Design (20%): Are operations broken down in a reasonable way (e.g. classes and methods)?
- Style (10%): Is the program **indented** properly? Do variables have **meaningful names**?
- Robustness (10%): Does the program handle erroneous or unexpected input gracefully?
- Documentation (10%): Do all program files begin with a **comment** that identifies the author, the course code, and the program date? Are all the classes, methods and data fields clearly **documented (comments)?** Are unclear parts of code **documented (comments)?** (Some items mentioned may not apply to some languages)

A program that does not compile (run) will get at most **50% of the possible points**.