CS 328 - Homework BONUS

Deadline

Due by 11:59 pm on **MONDAY, March 20, 2017**

Purpose

To encourage you to get some PHP practice before the Homework 6 deadline.

How to submit

Submit your files for this homework using `~st10/328submit` on nrs-projects, with a homework number of **56**.

Important notes

- These homework bonus points will be recorded in the Moodle gradebook in such a way that their effect is as if that many points were added to one of your homework scores.
- **DO NOT USE ANY CSS FRAMEWORKS FOR THESE PROBLEMS** -- one of this course's purposes is to provide you with practice with "plain" CSS3.
- **NOTE:** you are welcome to use `require_once` and `include_once` in your PHP documents! BUT when you use them in problems' documents, be sure to also SUBMIT copies of all files that you are requiring/including!
- Remember to follow the **CS 328 SQL and PL/SQL Coding Standards** as given in the CS 328 Homework 2 handout and the public course web site for all SQL and PL/SQL code.
- Make sure that you have executed the scripts `create-bks.sql` and `pop-bks.sql`, and that the bookstore tables are successfully created and populated.
- Remember to follow the **CS 328 HTML5 Coding Standards** as discussed in class and as given on the public course web site for all HTML5 documents.
- Remember to follow the **CS 328 CSS3 Coding Standards** as discussed in class and as given on the public course web site for all CSS3 that you write, also.
- And, now, also remember to follow the **CS 328 PHP Coding Standards** as discussed in class and as given on the public course web site for all PHP that you write, also!

Problem 1 - up to 20 homework bonus points

Let's start out with some PHP *without* any data tier involvement (no OCI, no Oracle, no databases yet).

You have seen several examples, now, of a PHP document that creates an initial form and also responds to that form (for example, `try-oracle.php` and `insert-dept.php`).

Create a PHP document `php-play.php`, formatted by `php-play.css`, that meets the following requirements:

- It includes your name and last modified date in its opening comment, **AND** the URL this can be run from. **It will be an automatic -10 penalty from your bonus if this URL does not display this page when I/the grader attempt to use it in a browser.**
- Its initial form should contain at least THREE **different** types of form controls and a submit button.
- Its response should VISIBLY do something based on those form control element(s).
  - (This could be as simple as, for example, summarizing the user's form responses, or the response could be something more involved.)
- Make sure you appropriately sanitize the inputs provided by the user.

Submit your files `php-play.php` and `php-play.css`.

Problem 2 - up to 20 homework bonus points

The purpose here is to get some more practice querying from the application tier.

Make a **new** copy of your `bks.css`, just in case you decide to modify it for this problem (since you do NOT want to do anything that would "mess up" previous homeworks' problems making use of earlier versions of `bks.css`).

Consider, again, the posted `try-oracle.php` from the Week 7 Lab.

Using this as a basis, create your own `bonus-test.php` that:

- includes your name and last modified date in its opening comment, **AND** the URL this can be run from. **It will be an automatic -10 penalty from your bonus if this URL does not display this page when I/the grader attempt to use it in a browser.**
- includes your name visibly somewhere in the `body` element,
- queries at least one of the tables created by `create-bks.sql`,
  - such that you project at least TWO columns, at least one of which is an aggregate function's result (result from `min`, `max`, `sum`, `avg`, or `count`)
  - ...and use `group by` so that your result has at least 4 rows

USE your `bks.css` to style this; if you need more formatting for `bonus-test.php`, EITHER:

- **IF** you think these changes might ALSO be useful for FUTURE documents, **modify** your `bks.css` accordingly,

OR

- **ALSO** make a `bonus-test.css` that you add using a `link` element placed **AFTER** those for `normalize.css` **and** `bks.css` in the `head` element of `bonus-test.php`.

Submit your resulting `bonus-test.php`, `bks.css`, and, if you created it, `bonus-test.css`.