**Assignment Instructions**

Homework 05 - Ajax

Read this entire homework description once through before proceeding.

**Required Reading**

* LyndaCampus jQuery Essentials
  + [http://www.lynda.com/jQuery-tutorials/essential-training/48370-2.html](http://www.lynda.com/jQuery-tutorials/essential-training/48370-2.html" \t "_blank)
  + Accessing Data with AJAX
* Textbook: “Objects” in Chapter 6
* jQuery AJAX Tutorial:
  + http://learn.jquery.com/ajax/
  + TOPICS: Key Concepts, jQuery’s Ajax-Related Methods, Ajax and Forms, Working with JSONP, Ajax Events

**Task**

Create a web page that makes use of an AJAX API that works with dates.  This API will be provided to you that can run on a Cloud9 PHP-enabled workspace.

* Copy the files “calculate.php” and “php.ini” into your HW6 workspace
  + Note: the php.ini file is necessary because the timezone is set inside it
* Create a small form that calculates the difference between two dates:
  + A div/span that will contain errors
  + Textbox for date #1
  + Textbox for date #2
  + A button or link to calculate the difference
  + A div/span that will contain the results
* The button, when clicked must make an AJAX call to the “calculate.php” file.  It must pass date #1’s value as the “date1” parameter, and date #2’s value as the “date2” parameter.  A successful call to the server should show the result as the total number of days between the two dates inside of the div/span.  Ensure that errors are not shown when a successful call to the server occurred.
* The date text boxes must be validated so that only valid dates can be entered, which is checked when the button is clicked (MM/DD/YYYY).  Instead of a popup, make the validation errors appear in the “errors” div/span.
* If the AJAX call to the server has an error, display a simple error message “An error occurred contacting the server” inside the “errors” div/span.
* Use styling (bootstrap, or your own css) to make the form look appealing.
* Use unobtrusive JavaScript only.  jQuery is allowed and encouraged.

**Extra credit for:**

* While you are typing the dates, each time the date textboxes change, have it send an AJAX call to the server to calculate the total, instead of having to press the button.  It should only make the call if the data is valid.  If the data is invalid (e.g. a bad date, or is empty), show the error messages in the “errors” div as usual, but have it clear the errors when valid data is shown.

**API Information**

**calculate.php - calculates the difference between two dates**

Parameters:

* date1 - the first date, must be in MM/DD/YYYY format
* date2 - the second

Results:

* JSON object with the following fields:
  + days - total number of days between the two dates
  + invert - 1 if it is a negative number, 0 if positive
  + y - number of years (not the total number, but partial)
  + m - number of months (not the total number, but partial)
  + d - number of days (not the total number, but partial)
  + h - number of hours (not the total number, but partial)
  + i - number of minutes (not the total number, but partial)
  + s - number of seconds (not the total number, but partial)

**Handing in Your Work**

* Provide, in the Sakai assignment submission area, a URL to your Cloud9 workspace containing the homework content.
* Explain in a short paragraphs how you took the data from the textboxes and prepared to send them to the server via the AJAX call.

**Grading Criteria**

* 9-10 points
  + The form functions without error when the PHP environment is properly running.  Only valid form data can be used to make the AJAX calls.  The form does not look sloppy and makes visual sense when viewed by the user.  AJAX errors are handled as expected and the user is given proper feedback.
* 7-8 points
  + The form functions without error when the PHP environment is properly running.  Validation may not be properly performed on the date values.  AJAX errors may not handled properly.  Some styling issues may be present.
* 5-6 points
  + Validation and AJAX error handling are not working, and AJAX calls may be made at inappropriate times.  Form is visually unappealing, sloppy, or is unstyled.
* +1 for extra credit
  + Totals are displayed whenever date text boxes are changed, rather than when pressing the button, but only when valid data is present.

**Supporting Materials**

* https://sakai.rutgers.edu/library/image/sakai/binary.gif [php.ini](https://sakai.rutgers.edu/access/content/attachment/4448be91-5475-44e3-b238-ff418df21e5f/Assignments%202/a0a5e9c2-4bfd-448a-9118-8422b0076031/php.ini" \t "_blank) (67 KB)
* https://sakai.rutgers.edu/library/image//sakai/generic.gif [calculate.php](https://sakai.rutgers.edu/access/content/attachment/4448be91-5475-44e3-b238-ff418df21e5f/Assignments%202/c1335fef-2208-4c83-b5b1-0bd754d06300/calculate.php" \t "_blank) (1 KB)