Advanced Web Development and Web Scraping
Spring 2020
Assignment #3 -- JavaScript

Note: To receive credit for this assignment, show your instructor in class or e-mail the assignment (screenshots of 1 and the HTML files for 2-5) to dancikg@easternct.edu with the Subject: CSC 301, HW #3.

For background on JavaScript, you should complete the JavaScript tutorial from **JS Home** to **JS Mistakes** at https://www.w3schools.com/js/default.asp but you can skip the following:

- JS Array Sort
- JS Array Iteration
- JS Dates, JS Date Formats, JS Date Get Methods, JS Date Set Methods
- JS Math, JS Random
- JS Switch
- JS Loop While
- JS Type Conversion, JS Bitwise, JS RegExp, JS Errors

1. Complete the following Exercises from w3schools.com (https://www.w3schools.com/js/js_exercises.asp)

    a. JavaScript Variables, Exercises 2, 4, 5
    b. JavaScript Functions, Exercises 3, 4
    c. JavaScript Objects, Exercises 1 – 3
    d. JavaScript Events, Exercises 1- 3
    e. JavaScript Arrays, Exercise 2-3

2. Create a webpage that uses JavaScript to store three web page URLs in an array, and uses a loop to display the links. The page should look as follows, with the header displayed using pure HTML and the links displayed dynamically from your JavaScript array. Note that the links must be active links (i.e., the URLs are displayed using the proper <a> … tags), and displayed in an ordered list.

    Some useful webpages are the following

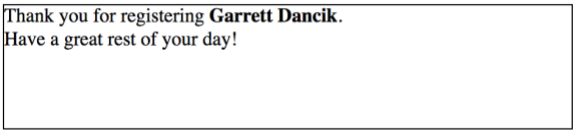
    1. http://www.easternct.edu
    2. http://gdancik.github.io/
    3. https://easternct.blackboard.com/

3. ***Multi-page registration site.*** Modify the *registration.html* page as described below. This assignment must be completed using JavaScript, without modifying any of the HTML code, unless stated otherwise.

    a. Use CSS to change the layout of the "page" div so that it is displayed on the top right of the "login" div as shown below: Hint: should the "page" div be "static", "relative", "fixed", or "absolute", with respect to the "login" div? Once set, where should its position be? Does the "login" div position need to be changed? **Note:** you must complete this question by changing the *position* and *location* CSS properties of the "login" and "page" divs only.

    b. When the user enters their first name and clicks the Continue button, they are taken to the next "page" and prompted to enter their last name. The page number, label, and placeholder are all updated by using JavaScript to change the HTML of the appropriate divs.

    c. After entering their last name, JavaScript is used to display a confirmation:



    d. If at any point the user clicks *Continue* without entering a name, an alert is issued that a name must be entered, and we stay on the current page.

4. Modify the web page you created for Assignment #2 as follows (you may simplify your web page if you like, but your table must have at least 4 rows and 3 columns containing a checkbox, a numeric input that accepts numbers in the range 1 – 10, and the item name. You may optionally choose to include tooltips and links as well. **Note:** the logic of your JavaScript code will require that each check box be linked to its corresponding numeric input (so that you can enable the first numeric input when the first checkbox is

checked, for example). One way to do this is to give each pair of inputs matching id numbers, e.g., “check1” and “num1”, and include the number as an argument of your JavaScript functions.

a. The numeric inputs should be *disabled* by default (simply include *disabled* inside the <input> tag, which sets the disabled attribute to *true*). The inputs should not display any value by default.

b. When the user checks a checkbox, the corresponding numeric input is enabled (use JavaScript to set the *disabled* attribute of the element to *false*) and given the value “1”.

c. When a checkbox is unchecked, the corresponding numeric input is disabled (set the *disabled* attribute to *true*), and any value in the numeric input is cleared.

d. When a numeric input loses focus (which triggers the *onblur* event), check whether or not the input is empty. If so, uncheck the corresponding checkbox and disable the numeric input.

e. When the user clicks the submit button, the form’s *onsubmit* event should call a JavaScript function which gives a summary of the order that includes the items and the quantity of each. This summary can be displayed in an *alert* or can be written to the window. As discussed previously, all inputs should be part of a single form, and invalid inputs should not be allowed (this will be the case if the form is set up correctly).

   Note: This should be handled by the form’s *onsubmit* event rather than by the *onclick* event of the submit button. The form’s *onsubmit* event handles invalid inputs based on the properties of its inputs (such as numbers outside the specified range). If the summary is handled by the button’s *onclick* event, then you would end up summarizing an invalid order. For example, the order would be summarized and then the form would display an error message such as “Please enter a value that is no more than 10”, which could be confusing to the user.

5. Modify the web page you created in the previous question, but the table should be created entirely using JavaScript (you may limit this table to 3 columns, as described in question 5). In order to do this, first create an array containing information about the items/products in your table, and use a loop to generate the HTML that will display the table. Your array should contain the item names (as strings), or if you want to display more information, then your array should contain *objects* containing the information to be displayed (e.g., item name, link, etc). Note that this framework makes it very easy to add items to the table, since the HTML code is automatically generated from the array.