CSE 391: Programming for the Internet

Assignment 2: JavaScript Programming

Deadline: As Per Lab Instructor

Note: The specific look and feel of the pages described below is left intentionally vague, allowing considerable design freedom on your part. However, for full points the pages should have a nice look and its code should satisfy common standards (meaning, for example, that it would be easy for someone else to read your HTML, CSS, and JavaScript code and **understand** what the intention of the pages is, how the JavaScript code is supposed to operate, and **it should be relatively straightforward to maintain**).

Submission of work
Please remember to have your web pages online so that I can access them via a web browser **and** to submit them to the mail by midnight before the day they are due.

Remarks:

1. There's loads of information about JavaScript that's available online, more than I can possibly tell you about in the lectures.
2. The third part of this assignment is likely the most difficult (so do the other two parts first).
3. You are definitely able to do all these tasks using only JavaScript (and HTML, CSS of course).
4. The functionality of your pages (at least for the second and third parts) relies on writing appropriate event handlers to control the tasks that you want to perform.
5. You will likely want to remind yourself/read more about the JavaScript method called "getElementById" which can be very useful to retrieve various elements of a webpage (using the ids that you assign to them). This then allows you to extract the values from those elements (like input text boxes, check boxes, radio buttions, etc) and/or assign new values to them. (See my JavaScript example where I use this method for an example.)
6. It's also possible to use the "innerhtml" property to change values (but *this can sometimes not work correctly based on particular browers or versions*, so read more about it if you choose to use this way to alter page contents). An alternative method to alter webpage contents dynamically is to use the methods of creating and appending (or deleting) the nodes of the Document Object Model.
7. Whatever method you choose, whether it's a straightforward method, or a more complicated approach (by manipulating the DOM), my suggestion is to proceed slowly and test your code a lot along the way.

Learn to debug your js with chrome dev tools or firefox firebug (http://getfirebug.com/).

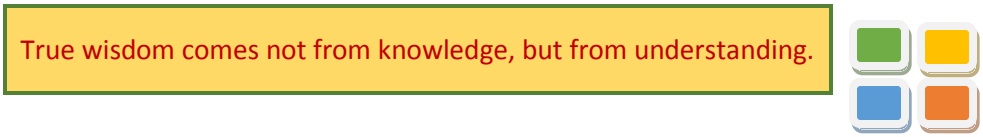
Use validator to quickly get syntax error:
http://validator.w3.org/
http://jigsaw.w3.org/css-validator/
http://webster.cs.washington.edu/jslint/

Use alert() and console.log() to print debug messages.

Part 1 (15 + 15 points)

1.1 Quote Box

Add a random "quote generator" to your home page. That is, your page should contain a list of fortunes (stored as an array of strings), and should randomly select one of those fortunes to display each time the page is loaded. The fortune should be displayed just above the page footer, centered and enclosed in a box. And there will be four color button beside the box. When we will click any color button the font color, the border color and the background color of the box will change. Also the font family and font size will change slightly. (Select your colors depending on your background. So that the color of the font, the border and background color of the box doesn't mix with the background color of the webpage.) Here's an example given below.

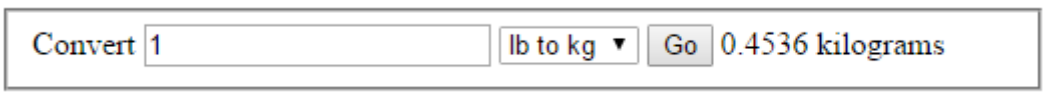


1.2 Hero Converter

Create html and write a function convert() in js file that takes the value in the text input and converts it from either from kilograms to pounds or from pounds to kilograms depending on the dropdown box. Here's an example given below.

The result should be displayed in the empty span.

The conversion factors:
1 pound = 0.4536 kilograms, and
1 kilogram = 2.2046 pounds.

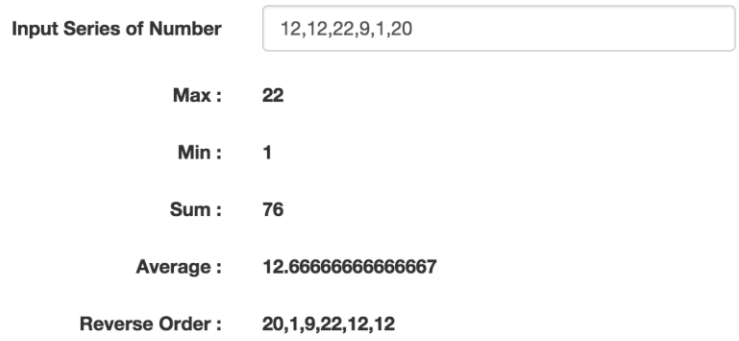


Part 2 (30 points)

You have to find the Max, Min, Sum, Average and reverse order of a series of no.

Create a webpage which will show the output (Max, Min, Sum, Average and Reverse Order) of a series of no. The page should include a form where an input field will be present a series of no. will be given input in the field separated by coma. The program will keep on doing the calculation as you keep on providing a series of no.

Sample:



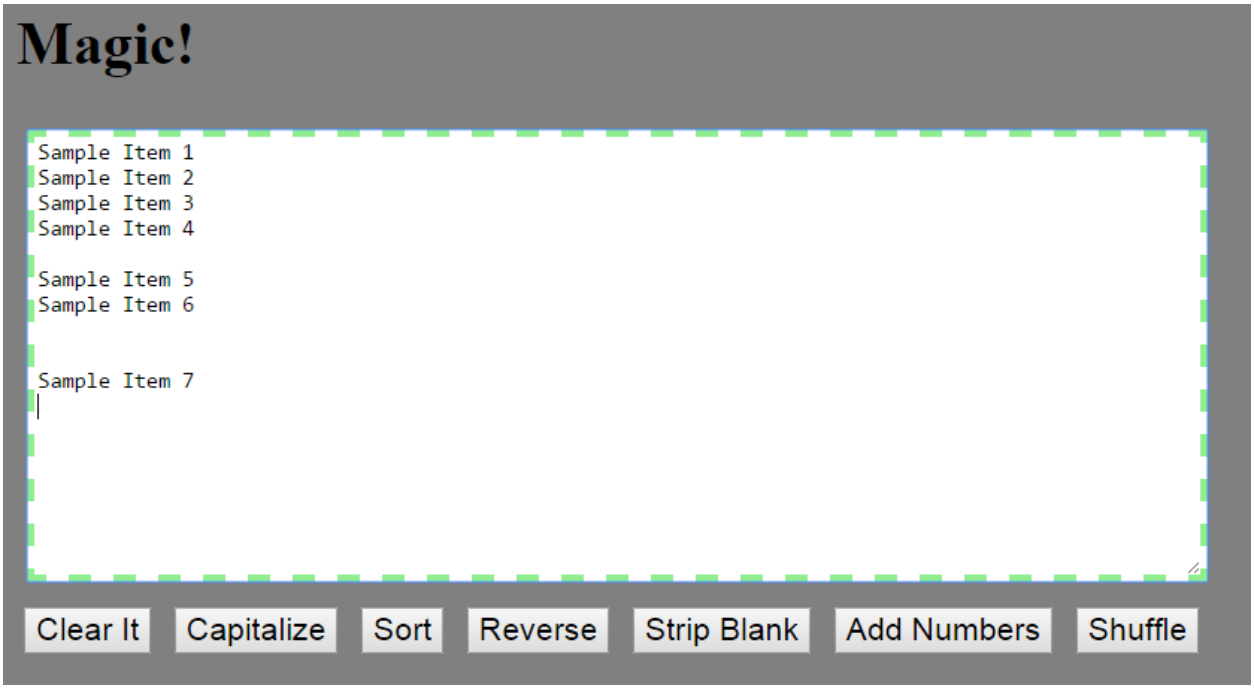
Part 3 (40 points)

Magic!:
The HTML structure is given below, we must add JavaScript code to make the UI respond when the user clicks the buttons.

Add the following behavior to the buttons:

Clear All: Deletes all text from the text area.
Capitalize: Converts the text to upper/lower case. Button will work as toggle.
Sort: Rearranges the lines into sorted alphabetical order.
Reverse: Reverses the order of the text in each line.
Strip Blank: Removes any empty lines from the text area. Also remove empty space at beginning or end of any line.
Add Numbers: Places a number in front of each line, such as "1. " (Ignore previous numbers in front of them.)
Shuffle: Rearranges the lines into a random order. Js do not have this built in, use Math.random().

Hints: Use getElementById to get String, which has a split method to break it into smaller string array. Then apply various native methods of array to get most of the things done.



Final mark

This assignment contributes 1/4 of the continuous assessment part of the course grade.