Assignment No. 6

Using the jQuery Validation Plugin with Your Dynamic Table

Date Due: ***Nov 12th, 2020***

What This Assignment Is About

As we have discussed, it's good for a GUI program to anticipate and compensate for user errors, but it's even better if the program is designed to catch errors as soon as the user makes them and guide the user toward correcting them. Remember the three pillars of good error messages:

- identify **where** the error occurred — direct the user to the precise **location** of the error
- identify **why** the error occurred — explain exactly **what** the problem is
- suggest **how to correct** the error — provide information to **guide** the user to correcting his/her entry

In our previous assignment, you were to do this using either "straight" JavaScript or jQuery by processing events such as `blur`, `focus`, and `submit`. For this assignment you are required to do the validation using the jQuery Validation plugin (http://plugins.jquery.com). This plugin will allow you to do more powerful validation and make your application more responsive to users. (As you know, jQuery is a JavaScript library built entirely in JavaScript, so it is *possible* to do everything that the jQuery Validation plugin does using "straight" JavaScript, but it is much, much more work.)

What You Are To Do

1. Begin by copying your program for the previous assignment — or at least your main HTML page — to a new file so that you can modify it without destroying your previous work.
2. Modify your page to validate all data entered by the user using the jQuery Validation plugin (http://plugins.jquery.com) or found here (https://jqueryvalidation.org/) and prevent the form from being submitted (or the table from being regenerated) if the user's entry contains an error. If there is no error, regenerate the table as in the previous assignment.
3. Explore the various jQuery Validation plugin options carefully. Read the text, study examples below, and search for additional examples online. Customize the error messages displayed by the plugin so that they make sense in the context of your application. Customize where the error messages are placed to maintain complete control over how your page looks.
4. Test your page thoroughly. Make sure that you handle each possibility.
5. Have a friend run your application and try to "break" it. Plug any "holes" that your friend finds in your application and/or its validation scheme.

Note: The nine sample programs can be found starting at:

https://jesseheines.com/~heines/91.461/91.461-2012-13f/461-lecs/code/jmh-table-v1.html

Submitting Your Assignment for Grading

Please submit on Blackboard:

1. Readme file:
    a. GitHub URL where your application resides
    b. Link to your Github repository
2. All related code as a zip

How You Will Be Graded

This assignment will be graded on a 20-point system with points awarded as follows. Please note that the lists of features provided below are not meant to be exhaustive. They are merely representative of the types of things we are looking for in each grading category. As always, 20% of your grade is for documentation.

Criteria (numbers in parentheses are the points that can be earned for that bullet item)	***Possible Points***
Program Integrity / Design • (4) validation is accomplished using the jQuery Validation plugin • (4) error messages are precise and relevant and helpful to the user • (4) error messages are positioned reasonably and direct the user to the error location • (2) error messages are styled reasonably • (2) page does not reload if there is an error in the entered parameters	16
Source Code Documentation and Formatting • user name and pertinent contact information appear in all source files • ***all*** files contain adequate explanatory documentation that is meaningful and does not merely echo code • ***all*** files are properly indented and formatted with adequate white space for readability • ***any sources used are cited in comments embedded within code*** • **any missing documents/link**	4

** This assignment is based on 91.461 by Prof. Heines.*