# JavaScript and jQuery Course Instructor Kris Secor

# Final Project

# How to Submit

You will submit an HTML document that includes the necessary JavaScript code. You will submit the JavaScript code in a separate file.

**URL (optional):** If you place your project on the web, all you need to do is to submit the correct URL.

You **must include** an HTML comment in the file immediately after the <!DOCTYPE html> statement. The comment must include your name and a description of the project.

# Grading

This assignment is worth 30 points.

2

|  |  |  |
| --- | --- | --- |
| **Item** | **Points** | **Description** |
| **File: name and identification** | | |
| 1. File name is correct or  File name is part of existing website or  Place the application on the web and submit the URL | 3 | The name of the HTML file must be index.html  This is so that both you and I can clearly identify which assignment the file is for.  If you submit the HTML file in a ZIP file, the name of the ZIP file must be yourname\_jsjqfinal.zip  If you split the JavaScript and/or CSS into separate files, the files must be named:  jsjqfinal.js jsjqfinal.css |
| 2. File includes an HTML comment with your name | 3 | You must include an HTML comment with your full name immediately after the <!DOCTYPE html> statement.  This is so that I can quickly identify who created the file. |
| 3. All variables are defined | 3 | Although JavaScript allows you to initialize and use a variable without defining it with var, that technique can lead to problems. In this course, all variables must be defined with var.  You only need to define a variable with var once, the first time it is used. |
| 4. All variable names are meaningful names and accurately reflect their usage | 3 | Creating meaningful variable names is one of the most important parts of coding. In addition to a meaningful name, the variable name should accurately reflect the purpose of the variable.  For example, a variable name of "sanDiego" is meaningful, but it is not accurate if it is used to represent the price per gallon. |
| 5. Statements and comments are clearly written and can be easily understood | 3 | There are many ways to write statements in JavaScript. The goal is to have statements that are correct and that can be easily understood. |
| a. Statements are clearly written and can be easily understood |  | You can organize the code any way that you want to, the sections in the starter file are only suggestions. |
| b. Comments are used to describe the code |  | You must include comments in this assignment. You should put a comment before each function and each major section of the code.  Each comment that is used to describe code must be meaningful and accurate. Comments do not necessarily need to be lengthy.  A comment should not just repeat what is being done in the code, the purpose of a comment is to describe *what* and *why* something is being done. |

3

|  |  |  |
| --- | --- | --- |
| **Item** | **Points** | **Description** |
| 6. The needed references to frameworks, JavaScript files, and CSS files are placed in the correct order. | 3 | The CSS files should always be placed in the <head> section and come before any JavaScript code. The jQuery framework must come before any JavaScript files that require that framework. The jQuery mobile framework requires the jQuery framework. |
| 7. The JavaScript is properly implemented. | 3 | The application should work properly in the context of the HTML web page in an obvious and useful manner. |
| 8. Code is well-formatted | 3 | Indent your JavaScript statements under the  <script> tag. Usually, you will indent statements by 2 to 4 spaces.  In this assignment, you will have several functions, and within the functions, you may have if blocks or other types of blocks. You must use correct indentation techniques for each block — statements that are inside the block must be indented.  You must align the opening brace { and closing } characters, to clearly indicate the beginning and ending of each block. There are several examples of correct alignment and formatting in the handout and slides for this session, and also in the sample files.  You should include a blank line as a separator between sections of code. For example, you can put a blank line after the variable initializations and before you start the calculations. Do not "mechanically" put a blank line after each statement unless it makes sense to do so.  The idea of using "whitespace" (blank lines) in code is to provide visual separation of important sections of code.  Be sure you are using a consistent formatting style for opening and closing brace ( { } ) characters. |
| 9. All other code (HTML, CSS) is well- formatted | 3 | You can lay out the page any way that you want, using HTML and CSS, or you can use simple HTML code that just displays the results.  Your HTML code should be well-formatted. Do not include extra blank statements where they are not needed, but use a blank statement to separate sections of the HTML. Be sure you have closing tags where they are required.  If you use CSS, the CSS must be well-formatted. Be sure you have correct { and } characters in your CSS. |
| 10. You code lints in jshint | 3 | Please make sure to lint your own code with jshint. |

4