Lab 4 – WEBD 6201

Objective:

This lab was made to practice and reinforce your jQuery skills along with learning how to implement jQuery plugins. You will have the opportunity to investigate and tweak an extremely useful jQuery plugin, which gives a basic HTML table element some robust functionality.

Notes:

- Submit your assignment before or on the deadline. Late submissions will be penalized with a 25% deduction.
- You will only be required to submit 2 files. Please make sure they are zipped and named STUDENTNUMBER_FIRSTNAME_LASTNAME_Lab4.zip, where STUDENTNUMBER is your actual student number, FIRSTNAME is your actual first name, and LASTNAME is your actual last name.
- You will be submitting the HTML code of Q1 and the HTML code for Q2. Any other included files will not be marked.
- Good luck ☺

1. For this question, you will be experimenting with a jQuery plugin useful for displaying, organizing and filtering large amounts of information.

    a. Open the *lab4_q1.html* file, take a look around, and make sure it works in your browser. You will be working with this file for this question.

    b. Once you open the file in your web browser, you will notice that there are a few headings, which are within a table element. You also should have noticed within your text editor that there is a large array of elements within the *users.js* file also provided to you.

    c. Import the *users.js* file within the <head> tag just like you would import any other JavaScript file. You will be using this *users* array for the rest of the exercise.

        - **Note:** The reason this array is stored as an external file is for modularity and clean code. Having this data stored separately will allow you to traverse and modify your code much easier.

    d. Import the jQuery library into your file within the <head> tag using the following link: https://cdnjs.cloudflare.com/ajax/libs/jquery/3.1.1/jquery.min.js

        - **Note:** Instead of downloading the entire jQuery library and importing it locally, it is a better option to use a dedicated Content Delivery Network (CDN) to do the heavy lifting for us. The above link is the CDN for the minified version of jQuery.
        - **Note:** The *filename.**min.js*** file extension is a convention used to distinguish minified JS libraries. Minified libraries are ones which are converted to have short variable names, less spaces, and generally more condensed code. It is better to use in situations where developers don't need to make any modifications to the library. If there was a situation where you would need to modify the actual jQuery code, you would have to download the un-minified library, modify it, minify it again for good practice, then import it locally.

    e. Using a *for loop* to iterate through every element within the *users* array, add a row with each user's *id, first name, last name, email,* and *gender* to the table within the <body> tag.

        - **Note:** There should be no modification to the HTML portion of this file, everything should be done in JavaScript and jQuery.
        - Your *for loop* will be in JavaScript, however you must use jQuery to insert the rows. You may use the following link: http://api.jquery.com/append/ to help you insert the user information as easily as possible using jQuery.

f. Once you are finished, save your file and open it with your web browser again to ensure it actually populated the table with all the user information.

  - Note: Don't worry if it takes a few seconds to load the page at the beginning – DOM manipulation is very expensive.

g. Import the jQuery based DataTables library into your file within the <head> tag using the following link: https://cdn.datatables.net/1.10.13/js/jquery.dataTables.min.js

  - The DataTables documentation can be found at: https://datatables.net/

h. Import the CSS based DataTables library into your file within the <head> tag AFTER the file you imported in the question above using the following link: https://cdn.datatables.net/1.10.13/css/jquery.dataTables.min.css

  - Note: This will help style the table to make it look like the example on the DataTables site.

i. Instantiate your DataTables plugin and apply it to the table you populated above.

  - Hint: Instantiating the DataTables AFTER you've populated the table with all the entries will save you a tremendous number of headaches. Try instantiating it before and after the population of the table and see what happens.

j. Save your file and run it within your browser. Have a look around and play with the filters at the top.

2. Create a copy of the file you created above and name it *lab4_q2.html*.

    a. Modify the DataTable to only allow filtering by ID, Last Name, and Gender.

    b. Create a form below the table, which allows the user to type in a First Name, Last Name, Email and Gender (the Gender should be a dropdown with only two options). Create a submit button for this form as well.

        i. This can be done by writing the HTML for it. No need to use JavaScript or jQuery to create these.

    c. Using jQuery, add an event listener for the submit button to be triggered on click.

    d. Once the user clicks submit, the page should do the following:

        i. Using a JavaScript function, validate that the email the user entered does not already exist in the current *users* array.
        ii. If the email exists, output an error under the form explaining that the user could not be added because the email already exists.
        iii. If the email does not exist, add all the information entered as a new user into the existing users array and output a success message under your form.

            - **Note:** You will have to create a new unique id for each user created.

        iv. Using jQuery, make sure that this user is also entered into the DataTable above. You may have to modify some code to accomplish this.

            - **Hint:** The following page will be extremely helpful for this task: https://datatables.net/examples/api/add_row.html
            - **Note:** The reason you have to manually add a row every time a user enters their information is because you manually populated the rows to begin with, so the DataTable is unaware you entered a new user into the array. You must explicitly enter a new row every time because of this.
            - **Note:** Be sure that your DataTable retains the user when the pages are changed.

Question	Marks
1	25
2	25
Total	50