10-31 Ajax Form Assignment

Overview:

Modify the existing form assignment such that it submits the data via an AJAX post. Upon a successful POST the application will receive json data which the application will then display.

API:

You will use the following API for POSTING your data.

1. url
    a. https://ceclnx01.cec.miamioh.edu/~campbest/cse383/forms1/form-ajax.php
2. Data Sent
    a. user: string
    b. affiliation: string
    c. comment: string
    d. car: string
    e. color: string
3. Result:
    a. status: "OK" or "FAIL"
    b. msg: String
    c. if OK, will also contain data[]
        i. data[]
            1. user
            2. affiliation
            3. comment
            4. car
            5. color
            6. timestamp

Note: This only responds to a POST request.

Instructions

1) Your HTML application will start with the form.
2) After validating all inputs (as you already should have done), override the submit so that it makes an JQUERY, ajax POST call to the above URL.
3) Upon success and if status == ok, hide the form and display the returned data as a table.
4) Upon an error, retry the jquery post for a maximum of 3 times. After 3 times display an error message to the user.
5) At the bottom of the page, place a button that will cause the form to be unhidden which will allow the user to enter new form data.
6) Make sure you html is valid, indented, and well commented
7) Details:
    a) You will do all work in your existing working copy of your repository. Last assignment had you put your code in a directory called form-jquery. Now copy over this code to form-ajax directory. Make sure to commit this code to your existing repository.
    b) You must place ALL javascript in an external script file.
    c) You still must validate the user input via jquery before calling submit.
    d) You must check for ajax post errors. I will randomly (about 50%) of the time cause my rest api to return errors. If you get an error you should retry the ajax post call.
8) Submit to canvas
    a) A link to your HTML page
    b) A link to your gitlab repository

comments:

1) Making the ajax call.

You need to make sure to set the content type of the sending ajax when making the call as well as making sure to stringify the data object.

Assume:
var dataFromForm = {};
dataFromForm.user = "scott".
….

(you would really read the data from the form field, I am just showing you an example).

Then part of the ajax call will look like:

```
$.ajax( {
        url: …,
        contentType: 'application/json',
        data: JSON.stringify(dataFromForm),
```