SE 319

Assignment 1 Report

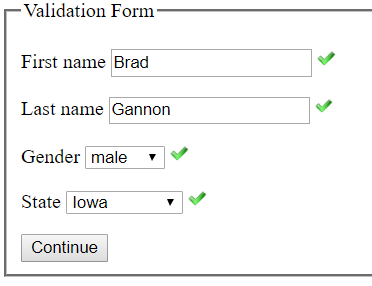
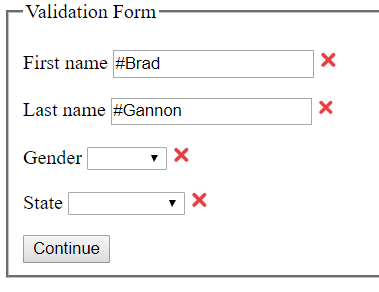
Brad Gannon

Validation 1 (Part 1)

The code I based my assignment on was given during Lab 2, and from there I simply had to edit and make small modifications to both the HTML and JavaScript portions. Starting with the HTML file, I had to rename it and create the text boxes that were correctly identified for the first name and last name. Next, I did some quick internet research to figure out how to make the dropdown boxes, which used <select> as the basis for organization. For both the state and gender selection areas, I included a blank option, which was then used for a Boolean in the JavaScript portion. At the very bottom of the HTML file, I then filled the *onClick* value with validate1(), which had the value of *continue*.

Moving to the JavaScript portion of Validation 1, I simply duplicated the given code in the validate1() function three times, so that each input section had a validation area associated with it. As far as the validation goes, the only thing I had to create was a quick function to make sure a blank answer wasn’t given for state or gender. All other necessary validation functions were given. Lastly, I used some Booleans to make sure the information was valid and added a line to move to Validation 2 if all information was entered correctly.

Here are screenshots of Validation 1, with the left image being correct entries and the right being incorrect entries. (I could not show in a screenshot that the *Continue* button works, but it will be shown in testing by TA).



Validation 2 (Part 2)

Again, this code was essentially taken from Lab 2 and modified to fit the specifications of this assignment. More specifically, I had it basically created from Validation 1, with only slight changes needed for completion. For the HTML file, all inputs were text boxes, which only required some copy and pasting and renaming the “id” values. The JavaScript portion required a little more editing and creation of some functions, especially for the phone number and addresses. The email validation function was already given from lab. I used a simple system for verifying a valid phone number, which essentially decided whether the input had hyphens or did not hyphens, and then checked if each value was a number. If the hyphens were included, a simple check was also added to verify they were in the correct locations. The address check was a bit trickier, as some city names can include hyphens and spaces. Therefore, I located the index of the comma and then checked if there were characters on the left that were either letters, hyphens, or spaces. However, there could be no more than two hyphens or spaces in a row, which would imply an invalid city name. For the state, I figured that a space could or could not be added after the comma, but then the last two values had to be letters. If more than two letters were present, it was assumed that the state abbreviation was invalid. I utilized a regex for checking numbers and letters. Here are screenshots of correct inputs (on the left) and incorrect inputs (on the right).

