WEB222 Assignment 5

Submission Deadline:

Sunday, Apr 9th, 2017 @ 11:59 PM

Assessment Weight:

5% of your final course Grade

Objective:

- To work with HTML form elements and CSS styling
- To perform client-side validation on user entered data.

Specification:

In this assignment you will implement the **HTML**, **CSS & JavaScript** code to recreate a functioning client-side implementation of a **professional quality Account Registration form** (see sample image to the right). It must match the sample as closely as possible and contain the form controls listed below under "**Form Controls**".

Upon submitting the form, it must adhere to the validation rules outlined below under "Validation Rules".

To get started:

- 1. Create an HTML5 document (assign5.html), a CSS3 file (css/assign5.css) and a JavaScript file (js/assign5.js) under a new folder.
- In assign5.html, create a form named "signup" to accept user's input for creating a user account. This form will submit to "https://httpbin.org/post" using the "post" method only when the form does not contain any validation errors. Lastly, set the title of the page to read "Sign Up".

Sample Image:



Click Here for a larger version

Please note:

- In order to help you achieve the required CSS to generate the form, you may make use of online tools such as those available on css3gen.com (ie: box-shadow, text-shadow & background gradient generators).
- elements are not permitted to be used in this assignment for positioning your elements
- Radio buttons, file and drop down (select) control appearance may differ from browser to browser

Form Controls

In the form "signup", the user input information will be grouped in to four sections: **Basic Information**, **Education**, **Work Experience** and **Attachments**. Add the following elements to each group in the form:

"Basic Information" Section

"Email"	Type of email , autofocus and required . User needs the email address for logging in the website.	
"Password"	Type of password, required.	
"Retype Password"	Type of password, required.	
"First Name"	Type of text , user needs to input his/her full name, required .	
"Last Name"	Type of text , user needs to input his/her full name, required .	
"Phone Number"	Type of tel , user needs to enter his/her phone number, required . This field should show the telephone input format using a placeholder	
"Address"	Type of text , home address, required .	
"City"	Type of text , city of the address, required .	
"Province"	Dropdown list : select element with options - the 10 provinces and 3 territories of Canada, required. The full names of provinces and territories will be showed up on the list and organized in two groups : Provinces and Territories . The values for option elements must use standard 2-letter abbreviations .	
"Country"	Type of text with aa datalist, required. Put five countries on the list: Canada, The United States and any other 3 countries in the world.	

"Education" Section

"School"	Type of text , the school that user/student is attending or the last attended, required .	
"Program"	Type of text , program in above school for the student, required .	
"Education Level"	Dropdown list: select and option elements, required. The Education Level list should contain the following items: - select -, Technical Diploma, College Diploma, Bachelor's Degree, Master's Degree and Others. The option tag for "- select -" should have a value of "0" or an empty string.	
"Education Status"	Type of radio . The value should choose from one of the following: Graduated , Current enrolled, Did not graduate.	
"Graduation Date"	Type of text , user needs to input his/her graduation date (any format), required .	

"Work Experience" Section

"Employer"	Type of text , the current or last employer of the student, optional .	
"Job Title"	Type of text , the job title of the student in the above company, optional .	
"Job Duty"	Type of textarea , description of job duties or responsibilities, optional .	
"Worked Year"	Type of number , the number of years for the student worked for his/her above employer, optional . The number user inputs will be in the range of 1 to 20 .	

"Attachments" Section

"Attachments"	Type of file, user can upload multiple files at a time, optional.
---------------	---

"Reset" & "Apply

"Reset"	Type of reset , user can reset the form to the default values by pressing this button.	
"Apply"	Type of submit ,	

All Fields & Controls

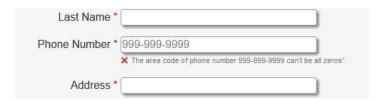
All form fields/controls should have **id** and **name** attributes as well as an associated **label** element if possible. Required field labels should be followed by a red asterisk (*).

Validation Rules

Some form fields require validation using JavaScript. The validation rules are as follows:

"First Name" and "Last Name"	 Must be present (cannot be empty string after trimmed) Allowable characters alphabetic characters lower case (a-z) upper case (A-Z) apostrophe (') hyphen (-) Must have at least 1 alphabetic character (a-z) (A-Z)
"Password" and "Retype Password"	 Password must contain both upper and lower case characters At least 1 number At least 8 characters long The value of the 2 fields must be identical unless one of them is an empty string.
"Phone Number"	 Must be present Must be in the format of 999-999-9999 ('9' here means digit only) The area code (first 3 numbers in 999) can't be all zeros (0)'s The actual phone number can't be all zeros (0)'s
"Education Status"	Must select one of these menu options

For the fields that require validation, your code should **validate the field** immediately **after** it has been changed (if applicable) (**hint**: use the **onchange** event handler to call JavaScript functions). Show **only one error for a field, at a time.** Appropriate messages must be shown for the invalid field. For example:



Other Requirements

- Your HTML & CSS must be well-formatted and properly commented where appropriate.
- Your code will be evaluated for correctness of syntax and appropriate usage.
- Each HTML page must pass the W3C's HTML Validation: https://validator.w3.org/
- All CSS used in the assignment must pass the W3C CSS Validation: https://jigsaw.w3.org/css-validator
- All JavaScript code must not contain any run-time errors/exceptions.

Final Submission Guidelines:

- a) Zip all of your files as assignment5.zip
- b) Upload the zip file to My.Seneca under Assignments -> Assignment 5 (same submission procedure as all previous assignments)
- c) IMPORTANT: All HTML & CSS Must not contain any errors when validated (HTML: https://validator.w3.org/, CSS: https://validator.w3.org/,
- d) **NO LATE SUBMISSIONS** for assignments. Late assignment submissions will not be accepted and will receive a grade of zero (0).
- e) After the end (11:59PM) on the due date, the assignment submission link on the Blackboard will no longer be available.