CS 6314 – Practice Work 2

Due date: September 18, 2022

Form validation is a way for you to tell the user what kind of input is required in each field and allows you to notify the user when the input is incorrect. There are many ways to do form validation (via HTML, client-side scripting or server-side scripting). In this assignment, you will use Javascript and write a plugin that displays information/error messages next to input fields.

Requirements

The code files for form validation are validate.html, validate.css and validate.js. You will only modify validate.js.

The validation rules are as follows:

- The username field must contain only alphanumeric characters.
- The password field should contain at least six characters
- The email field should be a valid email address (abc@def.xyz). A valid email consists of an email prefix and an email domain with two parts. <prefix>@<domain_part1>.<domain_part2>

When the page is fully loaded, you should insert a <span> notification element immediately after each form field. The notification elements should initially be hidden.

When the field is currently being edited, the notification element's text should be information message (i.e. password should include at least six characters), its class should be "info", and it should be visible.

When the field is not being edited:

- If the field is empty, the notification element should be hidden.
- If the field is non-empty and the form field validates, the notification element's text should be "OK", its class should be "ok", and it should be visible.
- If the field is non-empty and the form field does not validate, the notification element's text should be "Error", its class should be "error", and it should be visible.

Solution hint:
You should use the form events functions to detect when a form field gains or loses focus. You should not use mouse and keyboard events such as click or keydown: there are multiple ways to give a form field focus, and it is difficult to detect all of them reliably using mouse and keyboard events. Span elements can be added using element creation and DOM insertion.

For validating user input, you will need to write regular expressions.
https://javascript.info/regular-expressions

You can change css class of an element by using className attribute.

You can use *display* property in css to change an inline element to a block element.

Submission:
Zip your validate.js file and send it through eLearning. Do not include html and css files (remember you are not making any changes on those files).

Include screenshots for different cases (valid and invalid input)

Name of the zip file: PW2-yournetid.zip.