You create a JavaScript function that performs basic validation of username and password fields.

1. A common use of JavaScript is to perform form validation –

that is, look through input that a user has provided in a web form and make sure it follows the rules for input

First, fork and then clone the [ChicoState/BasicFormValidation](https://github.com/ChicoState/BasicFormValidation) repo.

Your cloned repo will have three project files: form.html, form.css, and form-validation.js.

**form-validation.js** contains all of the JavaScript used in this lab.

**form.html** has a bootstrap-style registration page with username (a text input with the id "user") and

password (a password input with the id "pass") and a button that, when clicked, invokes the JavaScript

function called **validate()**

Open ***form-validation.js code*** and read over the validate function.

It already performs a few actions:

* storing the user's input in two variables
* shows error messages
* changes the classes used to style the form to indicate an error.

However, it does all of these actions regardless of what the user types in.

For this lab, we want to validate user input using two rules:

* 1. The username must be at least 6 characters long
  2. The password cannot be "password"

Tackle these two rules one at a time.

First, the username typed in has been stored in the variable **userEntered**.

To enforce the rule, we need to figure out how long the user's entered username is.

Review the [methods for strings](http://www.w3schools.com/js/js_string_methods.asp) to recall how to get that value.

Once you identify that value, you need to make a [condition](http://www.w3schools.com/js/js_if_else.asp) that checks if the **value is at least 6**.

* 1. ***If it IS at least six***, no error should be shown for the username but instead of changing the usernameGroup form elements to red (by adding class "has-error"), let's change it to green by **adding the class "has-success"**.
  2. If it IS NOT at least six, use the code provided to show the error message and change the usernameGroup to red

Once you have this working, this would be a good time to stage your changes, commit them, and push your latest version to your GitHub remote repo.

Similarly, we want to ***add conditions that shows an error if the password entered is the word "password"*** but otherwise shows the password field with the class **"has-success"**.

Use what you learned from the first half of the lab to [compare the variable **passEntered** to **"password"**](http://www.w3schools.com/js/js_comparisons.asp).

 Make sure that the validate() function checks username and password **independently**; in other words, if the user entered a valid username but an invalid password, only the password field should show an error (as well as vice-versa).

Make sure to stage, commit, and push your final version. Submit by pasting your fork's URL as a Text Submission.