

Assignment 2: Login with Encryption & Testing

Due date: 11:59pm on Friday February 12, 2016.

Marks: 10

Format: Send zipped folder containing all files to sean.doyle@humber.ca.

BE SURE TO PUT YOUR NAME IN THE FILENAME: e.g. *Assign2-YourName.zip*

Purpose: To implement JavaScript encryption with CryptoJS, and unit testing of your functions with the Jasmine testing framework.

Add encryption to your login, and test functions with Jasmine.

REQUIREMENTS: Using your login code from Lab 1...

1. In addition to the onsubmit function, you will create two custom functions:
 - a. **md5Encrypt** (encrypt a string with MD5 and return a string)
 - b. **checkLogin** (validate user input)
2. The empty div should be used for a successful login message as well as error messages.
3. Incorporate Jasmine testing to test your two custom functions. Your spec file should test for the following specifications:
 - **md5Encrypt** should return a string 32 characters long.
 - **checkLogin** should return true if the username and the password match a known username and matching password.
 - **checkLogin** should return "Invalid Username or Password." if the username and password are not a known matching username and password.
 - **checkLogin** should return "No username entered." if the username is an empty string.
 - **checkLogin** should return "No password entered." if the password is an empty string.

If you need some hints, scroll down...

To Get Started...

- In your Assign2-YourName folder copy your lab 1 login page files and unpack your Jasmine-standalone zip file. On the JavaScript page inside the onload function, create three functions.
- Your **onsubmit** function will get the values from the form, then pass them as parameters to your **checkLogin** function. Your checkLogin function will call **md5Encrypt** when it compares the user input to your hard-coded encrypted password.
- Remember, if you are using <form> and not just grabbing the <input/> by id, you need to return false at the end of your onsubmit function so that you can output to the same page. Otherwise, the page will refresh on submit and your form values will be gone.

Your Functions...

- Paste your previous validation for empty fields into checkLogin. You will most likely need to modify it.
- In checkLogin, normally we would go to a database and check for a matching username/password there. For this assignment, check against a hard coded user/pass. e.g. *if (username === "yourname"*
- Leave your output logic until after the functions have run, then output to the html page.

Testing...

- Don't forget to include CryptoJS in the specRunner.html file just above the <!-- include source files here... →.
- For testing your encrypt function, use Jasmine's toMatch() matcher with a regular expression (a word character x 32).