

IMY 220

Practical 5:

jQuery & AJAX

Due: Friday 18 September @ 10:30

The submission instructions are available on ClickUP. Any deviation from these instructions will cause a 10% deduction from your mark.

Instructions

- Enhance the user experience of Assignment 2 by checking the validity of the email address and password with AJAX before logging in. The result of this will be that the user is only redirected to *login.php* if their credentials are correct and receive an error message on *index.html* if they are not.
- Download *index.html*, which includes a form for adding products and examples of added products, and *checkPassword.php*, which contains PHP code for validating user details. The files for this practical also include *style.css* for displaying the form correctly.
- Note: for the sake of this practical, *checkPassword.php* simply checks the user details against an array of email addresses and passwords. In a real scenario, such as your project, this check will require a database call.
- Include the newest version of the jQuery CDN and create a file called *script.js* and include it in *index.html*. All your code for this practical must be written inside this file.

Section 1: Verify Credentials

Inside *script.js*, use jQuery to accomplish the following:

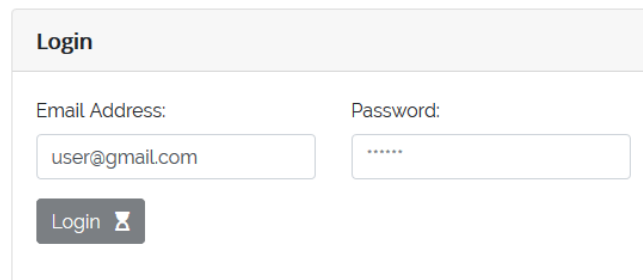
- When a user clicks the submit button for the login form, the form must be prevented from submitting.
- An AJAX call must be made that send the email address and password entered into the login form's input boxes to *checkPassword.php*.
- Only if the email and password entered must the form be submitted, thus redirecting the user to *login.php* as per the functionality for Assignment 2.
- *checkPassword.php* contains code to check if the given password does not match the given email address or if the given email address does not exist in the array. Use this functionality to display Bootstrap error messages for the appropriate error. For example:

The image shows two side-by-side screenshots of a login form titled "Login". Both forms have an "Email Address:" label and a "Password:" label. The left form shows the email address "wrongpassword@gmail.com" and a password field with six dots. Below the inputs is a "Login >" button. A yellow error message box at the bottom says "Password incorrect". The right form shows the email address "notindatabase@gmail.com" and a password field with three dots. Below the inputs is a "Login >" button. A yellow error message box at the bottom says "No user with that email address".

Bonus: Loading

To ensure a satisfying user experience, provide meaningful feedback to indicate the system status (<https://www.nngroup.com/articles/visibility-system-status/>) by informing the user that the page is processing their request while the AJAX call loads. This must be done as follows:

- When the login button is clicked, the button should be disabled. As soon as the AJAX call has finished, regardless of whether it was successful, the button should be enabled again. In other words, the button should only ever be disabled while the AJAX call is being processed.
- While the AJAX call is being processed, the right-angle icon should be changed to an hourglass, which should be displayed upright. When the AJAX call is finished, regardless of whether it was successful, the icon should be changed back to how it was originally, including the hover-over functionality. You will need to edit *style.css* and add/remove classes to accomplish this.
- (You can use the PHP *sleep()* function inside *checkPassword.php* to simulate waiting for the AJAX call to finish).
- The button, while waiting for the AJAX call to complete, should thus look like this:



The image shows a login form with a title 'Login'. It has two input fields: 'Email Address' with the value 'user@gmail.com' and 'Password' with six asterisks. Below the fields is a 'Login' button with an hourglass icon, indicating a loading state.

Additional Information

- Refer to the slides as well as jQuery documentation (<https://api.jquery.com/>) for help

Submit only the following file(s) according to the submission instructions.

- *index.html*
- *script.js*