

### Before you begin.

This test **MUST** be completed in our assigned classroom during the assigned period. Any students who attempt to complete the exam in another location will be committing academic dishonesty. Additionally, students may use their code created in this class during this term as well as any notes they have taken. You may **NOT** use other student's code, code from a previous term, or other student's notes.

You have made it through the first third of our class! As a celebration our local farmer has asked for your help completing a web application she started. The new application will be used by many users and is ideal for collecting details on animals. As much as the farmer fancies themselves a hero - they do not have the super power of web application development. Below is their punch list of items that needs to be completed on this assignment along with the reward for each one.

Note, this application has been setup as a **single page application**.

The documentation for the web services is available here: <https://simplefarm.thesimplehomestead.com/api/>

### Account Creation

All users must create an account before using the system. To create an account they will need to provide a first name, last name, email address, class, and a password. There are no requirements on the data other than it being at least 1 character and no more than 50 characters. (35pts total)

1. Add the necessary references to utilize **Bootstrap, jQuery, and Sweet Alerts** in the page. **Bootstrap** and **Sweet Alerts** should be included as **local** resources where as **jQuery** should be included via **CDN**. (2pts)
2. Set the **'type'** on all inputs to match the expected data type (4pts)
3. Edit the click function for the **Register Account** button to provide the following functionality:
  - a. Validate the user has entered a **first name, last name, email address, class, and password** for verification (4pts).
  - b. Ensure the entered **class** is **'CSC3100'**(2pts).
  - c. If any of the data validation fails, display an appropriate message to the user indicating what failed using a **SweetAlert**. (5pts)
  - d. If data validation passes, use the **users.php** web service to **create** a new account. (5pts)
  - e. On return from **users.php**, if the response is an error response display a message to the end user utilizing a **SweetAlert** indicating the feature did not work as expected. (2pts)
  - f. On return from the **users.php**, if the response is not an error, **create** a new session using the **sessions.php** web service. (5pts)
    1. On return from the **sessions.php**, if the response is an error response, display a message to the end user utilizing a **SweetAlert** indicating the feature did not work as expected and that they should return to login to complete logging in. (2pts)
    2. On return from the **sessions.php**, if the response is not an error, add the returned **SessionID** as a **sessionStorage** item with the key **'SimpleSession'**.
      1. Toggle the display of the Create User <div> and the dashboard <div> (4pts)

## Logging In

A user should be forced to login before using the application. On the login page, the user will be required to enter a valid username and password combination before being sent to the dashboard. If the user does not have an account, there must be the ability for them to navigate to a location to create one. (21pts total)

1. Set the '**type**' on the email and password input to match the expected data type (2pts)
2. Update the click event for all buttons that have the class of '**btnSwap**' to allow them to toggle the display property of the Login and Create User **<div>** elements. HINT: `slideToggle` (2pts)
3. Using **Bootstrap** classes, add classes to the body to ensure it takes up the **full height of the screen, has a display of flex, and includes the primary color** as the background (3pts)
4. Since the username and password fields do not have labels, please add a **placeholder** on the username field (johndoe@email.com) and password field (password) to give a hint for the end user. Also add a '**hidden**' label for both the username field and password field that a screen reader can use to assist those folks with **visual impairments**. (2pts)
5. Edit the click function for the Login button to provide the following functionality:
  - a. **Create** a new session using **sessions.php** web service. (5pts)
  - b. On return from the **sessions.php**, if the response is an **error** response display a message to the end user utilizing a **Sweet Alert** indicating their username and password combination was incorrect. (2pts)
  - c. On return from the **sessions.php**, if the response is **not an error**, add the returned **SessionID** as a **sessionStorage** item with the key '**SimpleSession**'. (4pts)
    1. Toggle the display of the login **<div>** and the dashboard **<div>**. (1pts)

## Dashboard

The dashboard will allow a user to update their session with the last date/time of activity and review the animals in the database (46pts total)

1. Add three options to the Select status element
  - a. The first option should display an **empty string** and have a corresponding value of '' (1pts)
  - b. The second option should display 'Pig' and have a corresponding value of 'Pig' (1pts)
  - c. The second option should display 'Dog' and have a corresponding value of 'Dog' (1pts)
2. Add an **<input>** element to the form that contains the **<select>**. The input should be contained in a **<div>** with a class of 'form-group' with a label similar in style to the other elements on the page. This **<input>** element should have an id of '**txtVariableSearch**' (3pts)
3. Edit the click event for the button with id of '**btnSearch**' (2pts)
  - a. Ensure the user has selected either a non-empty value from the **<select>** element OR entered text into '**txtVariableSearch**' (2pts)
  - b. If the user has not selected a value from the **<select>** and has not entered text in the **<input>**, display a **Sweet Alert** indicating this to the user. (2pts)
  - c. If the user has properly entered or selected their **AnimalType**, use the **animals.php** web service to retrieve or read data from the database about the animals. This function has been started for you and should be ready for you to programmatically set the SessionID and the Animal Type. (3pts)
    - i. If a user has selected an empty value from the **<select>** element, use the criteria entered in '**txtVariableSearch**' as the **AnimalType**, else use the value of the **<select>** element as the **AnimalType**. (2pts)

- d. On the return from the web service, you should use the **sessions.php** web service to **update** the last activity date/time for that session. (4pts)
  - e. If the return from **animals.php** function is blank or null, **clear** the **<tbody>** and then show a **Sweet Alert** indicating there were no animals of the given type found. (2pts)
  - f. If the return from **animals.php** function is an array of animals, **clear** the **<tbody>** of the table, **iterate** across the newly returned data array and **build/add rows** to the database with the information for each animal. (8pts)
4. Edit the click function for the Logout button:
- a. Using the **sessions.php** web service, **delete** the session from the database.(4pts)
  - b. Remove the **sessionStorage** with the key of **'SimpleSession'**(2pts)
  - c. Toggle the dashboard <div> and the login <div> (1pts)
5. **IMPORTANT:** When a user navigates to index.html, you should check if there is a sessionStorage entry for 'SimpleSession' (1pts)
- a. If an entry exists, use the **sessions.php** web service to retrieve details about the session. (4pts)
  - b. On a return from the web service, if the response is an **error** response, remove the **sessionStorage** entry for **'SimpleSession'** (2pts)
  - c. On a return from the web service, if the response is not an error response, toggle the <div> for dashboard and <div> for login to display the dashboard to the user as they already are logged in. (3pts)

There are a total of 102 points indicated in the instructions. You can make a maximum of a 100 on the exam but having a few extra points is similar to giving you a 2 pt bonus that you get to choose.