CS230 - Web Information Processing

Assignment 4

Assignment Release Date:	15-03-2021
Submission Due Date:	02-04-2021
Feedback Due Date (estimated):	16-04-2021 (for assignments that make Due Date)
Support Laboratories	Online Labs 06-07 (Two Weeks)
Total Mark:	15%

This Assignment is worth 15% of the Web Information Processing CA Component.

This is an open-book, graded assignment. You may use online resources for reference purposes only to help with the assignment. Please cite all references as comments in your submissions. You cannot directly reuse HTML/CSS/JS **solution code** from online sources. **You must not engage with another student, in person or electronically (phone, social media, etc.) to secure assistance with this assignment. If you do so you will receive an automatic fail (0%).** We will perform similarity checks on submitted assignments to check for collaborative efforts. A reasonable attempt at this assignment will gain you 15% of your continual assignment marks. It is possible to gain extra credit (up to a maximum of 5%) for this assignment.

Assignment 04 - Working with CRUD and Online Databases

You are required to develop two applications for maintaining a database of personal details for users of a hypothetical online store. The database, which should be called USERS (note capitalisation), should contain tables that record user personal information [Title, First Name(s)*, Surname*, Mobile*, Email Address*], and both home and shipping addresses [Address Line 1*, Address Line 2, Town*, County/City*, Eircode]. The fields marked * are required fields, i.e., they must contain values.

Your solutions should provide CRUD functionality for Creating, Searching (Retrieving), Updating, and Deleting user information stored in a relational database (MySQL or MariaDB database) (i) using a “back end” application (worth 10%) and (ii) using a modified “back end” together with an AJAX-driven “front end” application (worth 5%) demonstrating a full-stack solution.

The following section details specific interaction requirements for both parts of this assignment.

Assignment 04 - Requirements - (i) Back End Development (10%)

For this part of the assignment you first need to develop an application that demonstrates “back end” functionality, i.e. implement the CRUD activity described below using randomly created, or hard-crafted, data. These data need to be ingested into your solution database and should be accessed using your back end application, which may be either a PHP or NodeJS application. Typically,

will need to self install PHP or NodeJS and database software in order to complete this assignment. Note that PHP applications may be executed from the command line once PHP is installed.

You should clearly indicate the sections of your (PHP or NodeJS) program that implement the CRUD activities outlined below:

(i) For the C (create) activity you should demonstrate how to create the user record (personal and address), and add it to the database.

(ii) For the R (retrieve/search) activity, you should randomly select and return all users matching a supplied name.

(iii) For the U (Update) you should update three elements of a specified user record (Phone, Email, Title) and all or any of their Address data.

(iv) For the D (delete) activity, please delete all records for a user matching a combination of Email, Phone and Name.

Please note:

(i) For this assignment, you should use a relational database to store the information for the user information using your XAMPP installation (or an online accessible database). Please use node.js that has been installed locally or CLI (Command Line Interface) PHP that comes with the XAMPP installation. If you do not have XAMPP/node.js installed please do so for this assignment. If you cannot install these as you do not have a computer that facilitates installation please contact John Keating (john.keating@mu.ie). A video will be posted to Moodle demonstrating how to use online resources for this assignment if you do not have installation privileges.

(ii) You should implement best practice when it comes to relational database design for this assignment, ie, you may need multiple tables. You may need to created index or ID fields not specified in the brief above. Try not to have a single monolithic table that contains everything!

(iii) You may write functions to randomly create personal and address data and you may use the name generator functions provide for the Assignment 03 if you wish.

(iv) Titles should include Mx, Ms, Mr, Mrs, Miss, Dr or Other (specify).

(v) You may reuse the created (or auto-generated) address as the shipping address or add a different address.

(vi) For this assignment you only need to validate data on the Title field.

Please note for the first part of this assignment you do not need to write any “front end” code. All of the CRUD activities can be implemented within the solution application.

This means that you should be able to run the application as a console application; this application implements the CRUD activities and outputs the information to the console display. You need to provide a data dump from your SQL USERS database together with your solution application in order

that it may be test4d and corrected. Failure to supply the data dump will mean we cannot test your solution, and we will automatically deducted half of the marks available for this section of the assignment.

Assignment 04 - Requirements - (ii) Front End Development (5%)

For this part of the assignment you will need to (i) update your existing "back end" solution so that it is accessible via AJAX, and (ii) develop a minimal HTML/CSS/JS "front end" functionality that demonstrates consuming two of the CRUD activity components (Create and Retrieve) using AJAX.

You may use the pure JS or jQuery AJAX implementations for this "front-end" demonstration. There are no requirements to develop a complete set of HTML/CSS/JS CRUD "front-end" forms for this assignment.

It is sufficient to use hard-coded data to send to, and revive data from, your "back end" service. Application output demonstrating the two CRUD activities should be output to an in-page page console (as I have done in some demo lessons to date), and not to the Browser JS console.

Assignment 04 - Development Notes

Please adhere to the following development requirements:

1. You may not use RESTful frameworks (for example, Express.js, etc.) for this assignment. This assignment requires you to use core fundamental code for manipulating databases. You may refer to online resources such as W3schools, of course. You may use any of the code provided in the CS230 Lessons or Lectures. Please refer to the lessons on AJAX and the demonstrations of NodeJS services for the second part of the assignment.
2. You must comment your code (for "front end" and "back end"), clearly indicating, how your code implements the solution described above in the "Assignment 04 - Requirements (i) and (ii)" sections.

Please note that there are many sample (JS/PHP) solutions for implementing similar solutions (database) functionality available online. While it is fine to consult these, and accompanying articles, for references, you may not re-use code from these projects. Please cite your reference sources in your codebase. We will search and identify online coding solutions to similar problems for the purposes of checking against submitted solutions in instances where we have concerns about code originality.

Assignment 04 - Extra Credit (5%)

For an extra 5% credit you are required to develop a fully functional HTML/CSS/JS User Interface (UI) that supports the full CRUD activity. You may use any JS (or other) framework to implement the UI (Bootstrap, jQuery, jQuery UI, etc.).

IMPORTANT SUBMISSION DETAILS

Before submitting your assignment students should check that their solution works in Chrome and/or Firefox. Please indicate the Browser, Operating System (Linux/Windows/MacOS) and Browser version used for testing (as a comment in your submitted code). If you use an online IDE please clearly specify the IDE and provide a link where possible.

All work must be submitted via Moodle (see "Assignments" section for submission). Work submitted via other means will not be accepted unless you have prior arrangements with the Head Demonstrator (Behnam Faghih). All work MUST be submitted by the due-date deadline. Late submissions will not be accepted.

*The assignment submission is a zip file named "**assignment-04-xxxxxxxxx.zip**" (where "xxxxxxxxx" is your student id) containing solution files, e.g. named "**assignment-04.js**", "**assignment-04.php**", etc. together with any other resources used in the assignment solution. External CSS and Javascript files should be named "**assignment-04.css**" and "**assignment-04.js**", respectively. Please ensure that all external files use relative directory referencing, rather than hard-coding the files' location.*