CS230 - Web Information Processing

Assignment 5

Assignment Release Date:	12-04-2021
Submission Due Date:	23-04-2021
Feedback Due Date (estimated):	30-04-2021 (for assignments that make Due Date)
Support Laboratories	Online Labs 09 and 10 (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.

Assignment 05 - Working with Online (NoSQL) Databases

You are required to develop a solution for an online mobile phone store maintaining a database of:

1. Personal details for customers of a hypothetical online mobile-phone store. The database should contain entities that record customer 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.
2. Item details for mobile phones sold by the online store (from a random selection of phone items; 10 maximum) which contain the following phone details: [Manufacturer*, Model*, Price*]. The fields marked * are required fields, i.e. they must contain values.
3. Order details for customer purchases. Orders contain a list of items (phones) that have been purchased by customers. Customers may purchase more than one item in a single order. Customers may make multiple orders containing multiple items. The system should record item purchasers; or it should be possible to establish this information using queries.

Your online database should provide CRUD functionality for Creating, Searching (Retrieving), Updating, and Deleting customer, item, and order information from a MongoDB database.

There are no requirements to develop a HTML/CSS/JS “front-end” for this assignment. You only need to develop the ”back end” functionality that demonstrates the CRUD activity described above using randomly created, or hard-coded, data in your back end solution.

Assignment 05 - Requirements

You should include methods for the each of CRUD activities for the different entitles. For example, for the C (create) activity you should demonstrate how to create the customer document (personal and address), and add it to the database using a method `insertCustomer(<details>)`. For the R (retrieve/search) activity, you should randomly select a customer and output (not raw object) the details (log to console) using `findCustomer(<details>)`. For the U (Update) you should randomly select a customer and update three elements of their personal (phone, email, Title) and all or any of their address data using `updateCustomer(<details>)`. For the D (delete) activity, please delete all records for a customer matching a specified email, phone and name using `deleteCustomer(<details>)`. Similarly, you need to perform similar CRUD activities for Items and Orders entitles.

You are required to develop either a NodeJS (JS) or PHP application that implements the functionality outlined above with the following additional constraints:

1. For this assignment, you should use a MongoDB database to store the information for the user information using either a local installation (or an online MongoDB Cloud Atlas database available here https://account.mongodb.com/account/login). Please use NodeJS that has been installed locally or CLI (Command Line Interface) PHP that comes with the XAMPP installation. If you do not have PHP/XAMPP/NodeJS installed please do so for this assignment. You will also need to install the drivers for accessing MongoDB. 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.
2. Please note that you should implement best practice when it comes to NoSQL (document) database design for this assignment, i.e., you may choose to have normalised or de-normalised models, or a combination approach.
3. 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.
4. Titles should include Mx, Ms, Mr, Mrs, Miss, Dr or Other (specify).
5. You may reuse the created (or auto-generated) address as the shipping address or add a different address.
6. Your code should include a brief description for the database design (your data modeling approach) and the impact on your code development. This should be included as a comment at the bottom of your code submission.
7. For this assignment you do not need to validate data (Please note that ordinarily I would have this as a requirement but given the current situation it is not required).
8. For this assignment you do not need to generate online forms to collect and validate data sent to the database (Please note that ordinarily I would have this as a requirement but given the current situation it is not required).

Assignment 05 - Development Notes

Please adhere to the following development requirements:

1. You may not use RESTful frameworks (for example, ExpressJS, etc.) for this assignment. This assignment requires you to use core fundamental code for manipulating databases. You may refer to online resources such as MongoDB, W3schools, of course.
2. You must comment your code, clearly indicating, how your code implements the solution described above in the "Assignment 05 - Requirements" section.

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.

Please "hard-code" your authentication details into the database. If you are using MongoDB please allow access from anywhere (whitelist 0.0.0.0). We will need access to your database in order to correct your assignment. **Please note that a MongoDB Atlas database is preferable for this assignment, you must create an account unless you do not have internet access.**

Assignment 05 - 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-05-xxxxxxxxx.zip**" (where "xxxxxxxxx" is your student id) containing solution files, e.g. named "**assignment-05.js**", "**assignment-05.php**", etc. together with any other resources used in the assignment solution. Please include a dump of the data from your database (as a text file) names "**assignment-05.txt**". Please ensure that all external files use relative directory referencing, rather than hard-coding the files' location.*