Lab 1 JSON, Object Constructor, localStorage (individually done) - Due Jan 21, 2024 11:59 PM

COMP-4537-0 - Internet Software Architecture (x-list 202410)

Please post the hosted url of your assignment with https// at the comments section before hitting submit, otherwise your assignment will be considered not hosted!

Learning outcome: Practicing serializing an object using JSON and storing in HTML5 LocalStorage and retrieving and parsing it in another page

Part 1 (8 marks) , implementation and deployment (hosting)

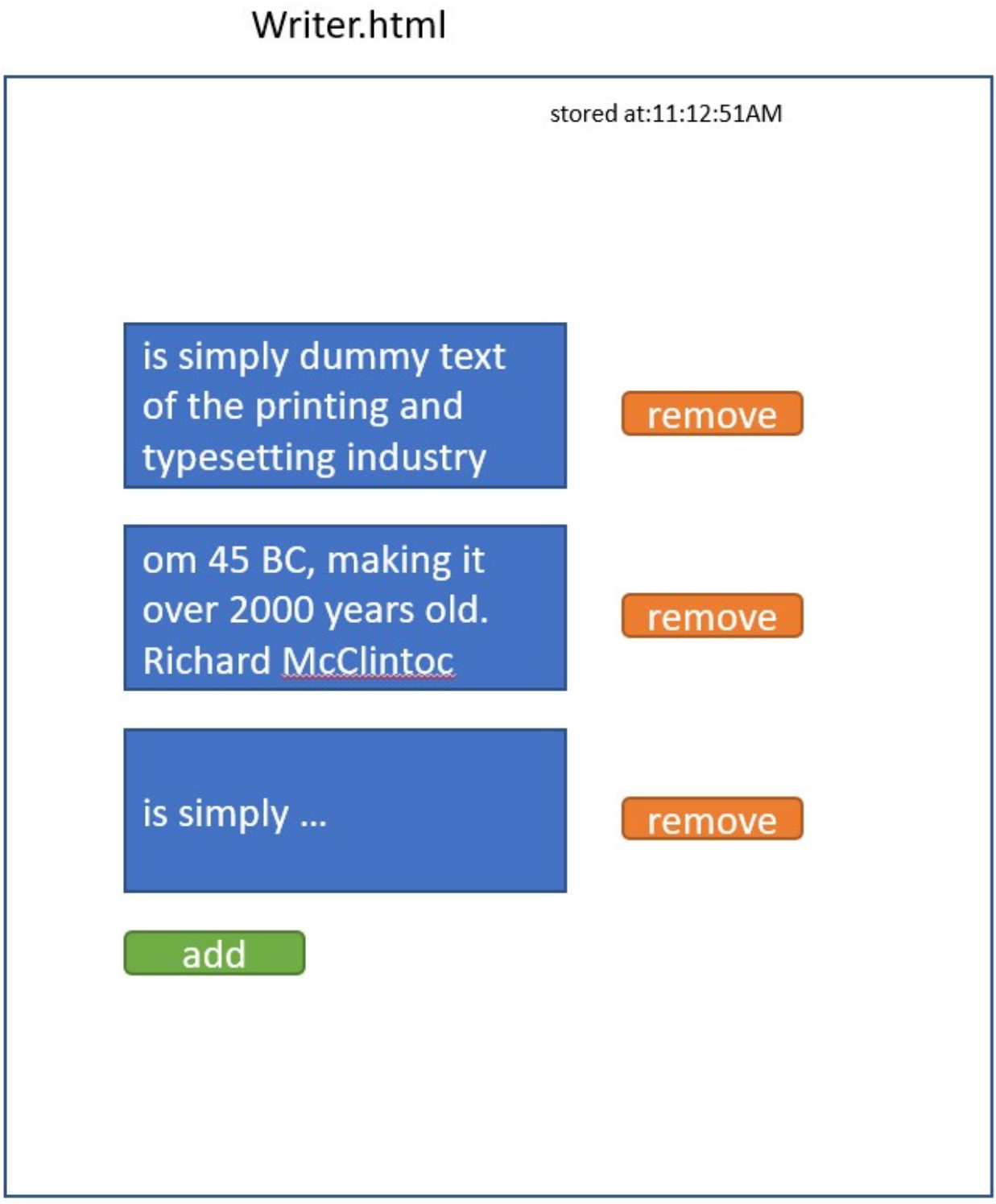
Implement a very simple note taking web app that uses the HTML5 local storage to store the notes in JSON format. The app is composed of **three pages**.

index.html which is the landing page of this lab **which has the title at top of your page (body of your page so that TA can see)**

Lab 1: JSON, Object Constructor, localStorage
e.g. Student name

with two labels (or buttons) hyperlinked to the next two pages of (writer) writer.html and (reader) reader.html. The two pages are as follows:

writer.html page, is for adding, removing or modifying the notes. It also stores the notes which are an array of objects into the local storage every 2 seconds*. It also constantly updates* and displays the most recent time at which the notes were saved, at the top right corner of the screen. e.g.

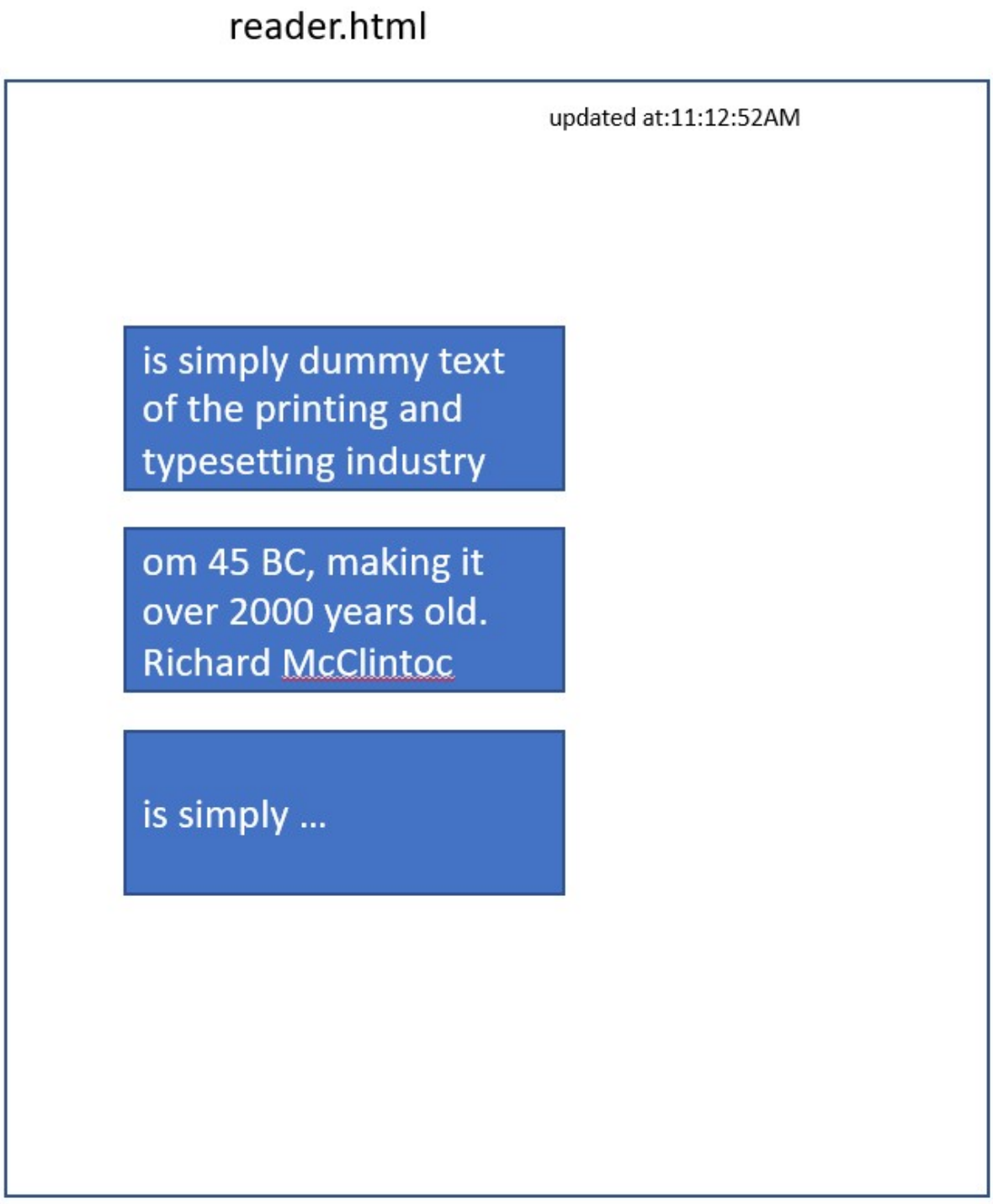


* some of you may be able to do that even more efficiently . i.e. is writing into local storage every 2 seconds necessary ?

note: pressing the remove button shall remove the corresponding textarea and its contents from the local storage instantly.
Pressing the add button adds an empty textarea and a new remove button next to it and pushes the add button down.

reader.html page is for retrieving the notes that the writer has created. It retrieves the notes every 2 seconds and displays a message at the top right

corner of the screen indicating the last time these notes have been retrieved. e.g.



Note:

You don't need to exactly replicate the UI styling presented in this screenshot .

Note:

At the bottom of both writer.html and reader.html there must be a back button bringing the user to the root of this assignment (the index.html)

Upon opening the write.html, retrieve the already existing notes in the localstorage and prepopulate the text areas with those. Say there are two notes already stored in the localstorage, then you would retrieve them from the localstorage once you open writer page and create two text areas as explained above and fill them with those notes(this way the writer user can edit/remove already existing notes)

Remember, if you open two tabs, one for write page and one for reader page. The moment that you update the content using the write page, the reader page also has to be updated automatically (without having the user to refresh the pae

Libraries allowed

Pure JavaScript (also referred as vanilla JavaScript) , Bootstrap, HTML, HTML5 and CSS and CSS3.

store the messages you display to user in a separate file \lang\messages\en\user.js

Directories/urls (its ok if you cannot exactly follow this structure**)**

At the bottom of both writer.html and reader.html there must be a back button bringing the user to the root of lab1 assignment which is https://yourDomainName.ca/COMP4537/labs/1/index.html)

Note: you don't need to have your own domain name. The root of your app could be something like https://myAdvWebDev.herokuapp.com

Part 2 Questions (2 marks)

1- open the writer.html and reader.html in two different tabs of same browser. Would the reader.html have access to the content stored in the local storage by the writer.html?

2- open the writer.html and reader.html in two different tabs of two different browsers (e.g. one in chrome, one in Firefox) . Would the reader.html have access to the content stored in the local storage by the writer.html?

Deliverable:

zip your Lab 1 files from the root directory and put them all in one zip file: yourLastNameLab1.zip e.g. GreenLab1.zip

together with your url (the root of lab1 url with https at comment section
and also answer to the two questions (put those answers at the comment section too)
e.g.
1- yes or no ...
2-..

Rubrics (Deductions)

(-4) if not hosted

(-4) for not using Object Constructors (or classes if you self studied) to form array of objects. e.g. you cannot predefine 5 variables and say your script can create maximum of 5 notes. Creation of textareas also has to be done dynamically (for example you cannot create 5 empty textarea when the page loads)

Additional deductions

-2 if you do not follow OOP and not use object constructor or classes in your code to create buttons

(-1 to -3) If Js, CSS and HTML are not separated.

.i.e. Your JavaScript code has to be in separate js files (no JavaScript coding inside your HTML file), CSS(if applicable) also in its separate CSS file (only external styling is allowed), otherwise marks will be deducted (of course sometimes you need ot change styling dynamically inside js, that one is ok)

(-1) for any extra file (e.g. none of these acceptable: this file was here from PHP, or node and I forgot to delete it, was put by IDE etc)

(-1) for any server side scripts (you don't need any server side script for this)

-10% for each day late

(-1 to -4) for not posting the url of your hosted assignment with https// at the comments section before hitting submit, otherwise your assignment will be considered not hosted!

-1 if you use http instead of httpS

if you are using https! with 's'. No warning security messages must be observed by the visitor of your website otherwise you will lose 4 points for not hosting properly

(-1) For wrong variable declaration. You cannot use var! use const if you don't need to change the variable's value later, otherwise use let for variable declaration.

(-1) If a variable is not supposed to be global, do not declare it global.

(-1) Do not use hard coded strings* all over the place in your code. Move all user facing strings p and declare them with const string variables and use those variables in your code instead .

Its ok to use chatGPT in assignments and term project provided
- You disclose using chatGPT inside your code (js file etc), otherwise -2 deduction
- and at the learning hub comment section when submitting your assignment otherwise -2 deduction
It is mandatory for you to comprehend each line of the code you have submitted. You will be questioned about it during marking which is done in-person, and for every incorrect answer, 1 mark will be deducted.

---------- even more tips

Your lab1 code has to be modular. You would need to use object constructor (older javascript) or class (new javascript style) .
e.g.

So you want to create an object constructor that creates your notes (every time you instantiate from it).
Depending on your design, for each note, you would need a text area and a button (e.g. remove) which are the properties of your object.
For each object you also need methods. Say a method remove(), which would remove the corresponding textArea and the button from DOM and also removes the corresponding content from webStorage.
Also may be a method to store the content of that text area into the webstorage
So we want to encapsulate the properties and methods of a note into a single object, so that each note has its own text area , its own button and its own methods (simple OOP concept).

There is an example of how to use object constructors (old js oop) in your second week lecture note, "COMP4537-2 Architecture JSON LocalStorage.pptx". Note that its not an answer to your lab1, its just an example.