Unit Project #2b

CSCI E-3 Introduction to Web Development Using Javascript
Harvard Extension School

This assignment is worth half of the total point value for the Unit 2 homework. The other half was in Homework #2a, previously distributed as a ZIP file.

What you're learning:

This assignment is designed to give you a chance to exercise your knowledge of objects and functions, creating and reading from a JSON data structure and an array, string handling, simple Web page interactions (reading from a form and writing to the page), and introduce the window.localStorage object.

What you need to do:

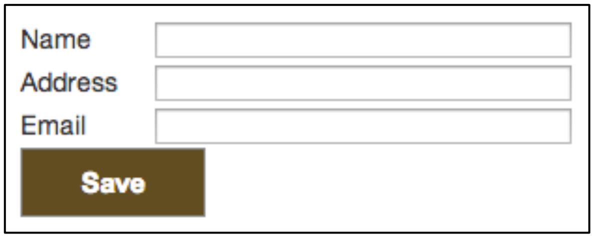
Requirement #1: For this assignment, you will write an HTML form that allows your user to input data of some type. It should contain at least three data fields that represent anything that interests you. For example, an individual record could be:

- A contact list that collects the name, phone number and email of your friends
- A movie list of your favorite films, including title, starring actors/actresses, duration, and year released
- A list of important places, including the name, a description, and geographic coordinates
- A list of the players on your favorite sports team, along with their current career stats

You get the idea - anything that interests you.

Your form will collect only one record at a time. So if you're collecting a contact list, your form will only have the input fields for a single person's information, along with a button to submit the information.

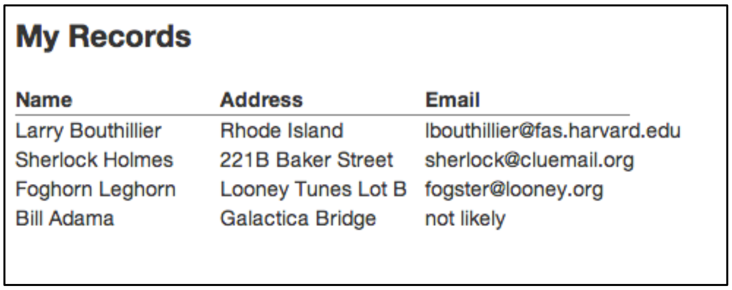
It might look something like this:



Requirement #2: When the submit button is clicked, two things will happen:

1. You will add the information to a list visible on the page. It does not have to be beautiful-looking, but it should be easy for users to read and understand what they are looking at.

   Your list doesn't need to be fancy. Here's an example:



My Records

Name	Address	Email
Larry Bouthillier	Rhode Island	lbouthillier@fas.harvard.edu
Sherlock Holmes	221B Baker Street	sherlock@cluemail.org
Foghorn Leghorn	Looney Tunes Lot B	fogster@looney.org
Bill Adama	Galactica Bridge	not likely

2. You will save the data in a JSON object and add it to an array, which you will store in window.localStorage. In an actual server-based web application, this array of JSON data might be sent up to a server for processing, but we'll just save it locally for simplicity.

Requirement #3: There's one more thing: When the page is first loaded, you should check for any data in localStorage and write that to the screen, so that a user always sees the entire list when they visit the page.

It is not necessary to create the ability to delete or modify existing records. We will be doing such things later in the course, but for now, it's out of scope for this assignment.

Some hints:

There's a lot going on here, but if you break it down into smaller chunks, it won't seem quite so bad. Here's some hints for how to think about your solution. This is *one possible* solution breakdown – you could construct your code and objects in varying ways to get this done. ***You may not use any Javascript libraries*** *for this, however. This specifically is designed to exercise your knowledge of the Javascript language itself.*

After (and during!) each step here, you should be testing your code and seeing that you're getting closer, step by step. Plan on lots of incremental iterations that bring you closer. Don't try to solve everything at once, and don't be surprised if you get a portion of the code working, only to have to change that code later as you continue to other requirements of the project.

1. **Create your data structure:** Your main data structure will be a JSON object that contains an array of other JSON objects. You can create these first by initializing an empty array, as well as a JSON object that contains one key (named as you choose), and the new array as its value.

2. **Build your HTML form:** Next, you need to create a form with several input fields, and a button to Save the record. Add the appropriate ids as needed so you can get to everything you need from Javascript. It does not have to use the HTML <form> tag, but it can if you like.

3. **Build a data object that will contain the data from your form:** If your form has three fields, you'll need an object that can contain these three properties. You'll create one of these every time you read data in from

the form or for each record from localStorage. For example, for my address book example, my object started out like this:

```
function AddrBookEntry(n, a, e){

          this.name=n;
          this.addr=a;
          this.email=e;
  }
```

To make a new record object, I can call the constructor

```
new AddrBookEntry(name, address, email);
```

We'll be adding more functionality to the object as we go, but this is a start.

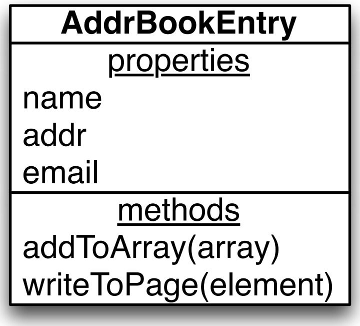
4. **Create an onclick function for your button:** In your JS file, you'll have to write an onclick handler for the Save button that does several things:
    a. **Get the values** from each form field
    b. **Create a new object** (using your constructor from step 3) that will contain the data from the form.
    c. **Push the new object into your array** from step #1. One way to do this would be to add an addToArray() method in your data object that accepts your array from step #1 as a parameter and then uses Array.push() to add itself to the array.
    d. **Add the information to your page** in an appealing way. Your page will need an HTML DIV or some other element you'll be writing to. You get the element using document.getElementById(), and pass that element to the method you'll write in the next step.

5. **Write data to the page:** Write a method, writeToPage(element) in your data object that that accepts an HTMLElement. Your method should write out HTML that formats the object's data for the page.

    Hint: you'll find that `element.innerHTML += [yourNewHTMLToAdd]` works pretty well for adding HTML to an existing DIV or other element.

6. **Store the data:** Now we need to store that information. Write another function that takes as an argument the JSON object that contains your array, and writes it to window.localStorage. Remember that localStorage only stores strings, so you'll have to call JSON.stringify() on your JSON object before you assign it to your localStorage.

At this point, you should have a program that lets you add information to the screen via a form in the Web page, and stores this information in localStorage. In my example case, the data object has the following structure:



The only thing left is to write code that will display any existing data in localStorage as soon as the page is loaded. So,

7. **Read from storage:** Add code that runs when the web page loads that:
    a. Checks to see if your localStorage data already exists for this page
    b. If so, recovers the JSON object by calling JSON.parse() on the data from localStorage
    c. Gets the array from the JSON object. This should live under the key you set up in way back in step #1
    d. Iterates over that array, creating the object you defined in step 3 each time, and calling its writeToPage() method.

Be sure you clear out your localStorage once in a while when you've gotten this working, to be sure it all still works when the localStorage starts out empty. You can do this from the Firebug console by running:

```
window.localStorage.clear()
```

You'll submit your assignment in the usual way, with one small change. Please name the folder you zip up like `lastname_firstname.hw2b` (*your* first and last name, please!). You should also create a ZIP file named like: `lastname_firstname.hw2b.zip`. You should use the same folder structure we've been using all along, with a conspicuous HTML file in the top level that runs your program.