Worksheet 11

**Movie Review Storage**

# What is HTML5 Local Storage?

1. Read: [http://www.w3schools.com/ html/html5\_webstorage.asp](http://www.w3schools.com/html/html5_webstorage.asp)

* Explain the difference between cookies and local storage?

|  |
| --- |
|  |

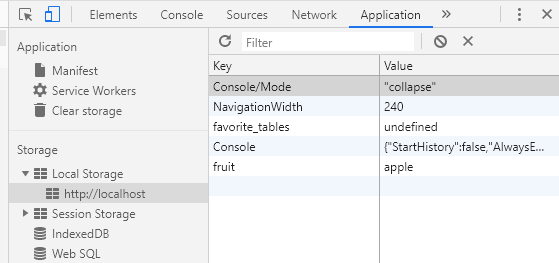
* When should local storage be used? See <https://css-tricks.com/localstorage-examples/>

|  |
| --- |
|  |

1. Where can you find the local storage in your browser?

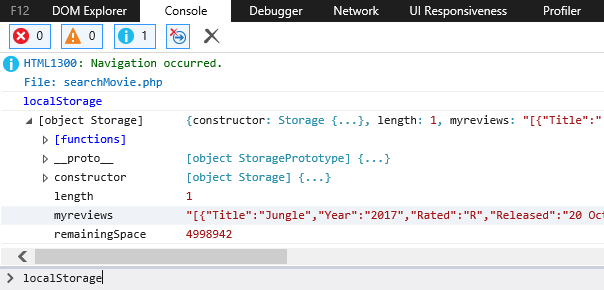
Chrome

1. Ctrl-Shift-I to display Developer Tools
2. Go to Application and expand Local Storage



Internet Explorer

1. F12 to display Developer Tools
2. Go to Console and type “localStorage”



# Setup

1. Download *C273\_L11MovieReviewStorage.zip* from LEO. Save and extract it to C:\xampp\htdocs\C273\_P11MovieReviewStorage\.
2. Start NetBeans. Create a PHP Project with Existing Source that points to above folder.
3. Start the XAMPP Control Panel, and **start both the Apache and MySQL component**.

# Store, retrieve and remove from local storage

#### In this section of the worksheet, we will see what are the methods to store, retrieve and remove values from local storage.

##### Resource: <http://www.w3schools.com/html/html5_webstorage.asp>

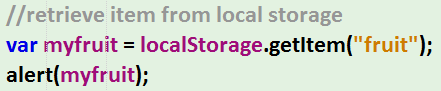
1. Create a new HTML file called exercise1.html. Type the code below within the <script> tags and run the file.



1. In Chrome, navigate to Developer Tools > Application > Local Storage under http://localhost . What do you think the code that you wrote in the previous step does?

|  |
| --- |
|  |

1. Continue to type the following code in exercise1.html to retrieve the value of “fruit” from localStorage. Run it and write down your observations.



|  |
| --- |
|  |

1. Continue to type the following code. Run it and examine the local storage in Chrome. What do you think this code does?

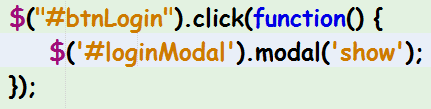


|  |
| --- |
|  |

# Retrieve values from form fields and store into local storage

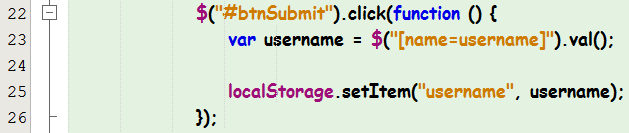
#### In this section of the worksheet, we will see how we can store values retrieved from HTML form fields into local storage. After which, we will read from local storage and display those values in the form fields.

1. Open exercise2.html. Type the code below inside $(document).ready(function() {…}); to display the modal when the Login button is clicked. Run the file to test it.



1. Run the file again, click on the Login button and enter some values in the fields. Currently upon clicking on the submit button, nothing is saved into local storage.

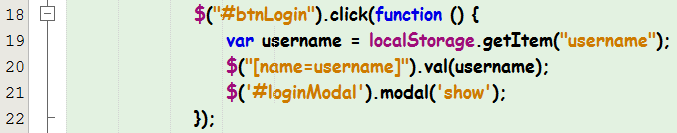
Continue adding the code below to exercise2.html Run it and check the contents in the localStorage in Chrome.



Explain what does each line of code do.

|  |
| --- |
|  |

1. Add the code below. Run it, click on the “Login” button and observe that happens.



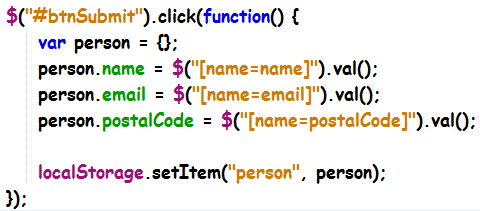
Explain what does Lines 19 and 20 do?

|  |
| --- |
|  |

# Store objects into local storage

#### What if we have multiple pieces of data related to each other which we want to store in a single key in local storage? This section of the worksheet will investigate how this is done.

1. Open exercise3.js. Add the code below to create a “person” object and retrieve the name, email and postalCode from the form fields and store them into the object. See <http://www.w3schools.com/js/js_objects.asp> on how to create JavaScript objects.



1. Run exercise3.html and upon entering the name, email and postalCode, click on the submit button. You should not see any output on your browser. Go see the localStorage in Chrome. Do you see the key “person” being stored in local storage? If no, why do you think that is so? Hint: <http://allthingscraig.com/blog/2013/01/07/using-html5-local-storage-to-store-objects/>

|  |
| --- |
|  |

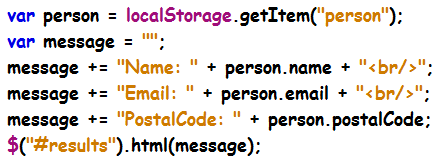
1. Using the online resource in the previous step, what is the method used to allow the storing of objects into local storage? What does this method do?

|  |
| --- |
|  |

1. Amend the code in Step 13 to allow the storing of the person object into local storage. Copy and paste the “value” generated in local storage. How is this different from the output generated by the code written in Step 6?

|  |
| --- |
|  |

1. Add the code below in exercise3.js.



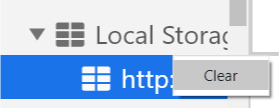
1. Run exercise3.html again. Do you see the name, email and postal code being displayed below the form? What causes the output is be displayed this way?

|  |
| --- |
|  |

1. Using the online resource in Step 14, what is the method used to allow the retrieving of objects from local storage for display? What does this method do?

|  |
| --- |
|  |

1. Amend the code in Step 17 using the method you proposed in the previous step to allow the display of the person details. Test your code.
2. Right click on <http://localhost> under Local Storage in Chrome and click on Clear at the popup.



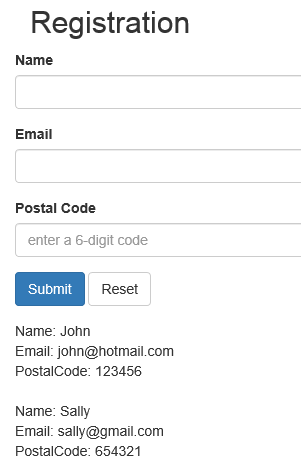
This will clear the contents in local storage. Reload exercise3.html again? What is the error message displayed in the Console? How is this error message related to clearing the local storage? How do you think you should fix this?

|  |
| --- |
|  |

1. Put in the changes that you proposed to fix the error in the previous step. Run exercise3.html and test thoroughly.

# Store list of objects into local storage

#### In the previous section, we were only able to store and display one person details from local storage. What if we wish to store and display multiple persons details and produce an output like below?



Before you continue, you may want to recap on JavaScript arrays from this resource: <https://www.w3schools.com/js/js_arrays.asp>

1. Duplicate exercise3.html into another file named exercise3\_list.html. Do the same for exercise3.js into file named exercise3\_list.js in the js/ folder. Remember to change the link in exercise3\_list.html to point to exercise3\_list.js.
2. Open exercise3\_list.js. Add code into the click event handler to store the person object into an array before storing into local storage as a JSON string. Make sure that the next person object can be added on to the same key in local storage.

The local storage content should look like this:

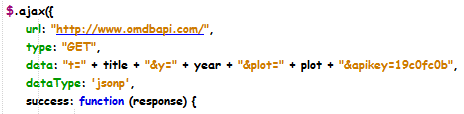
"[{"name":"John","email":"john@hotmail.com","postalCode":"123456"},{"name":"Sally","email":"sally@gmail.com","postalCode":"654321"}]"

1. Edit the code similar in Step 17 to process the array read from the key you have stored in local storage in the previous step. (i.e. Use for loop)

1. Run exercise3\_list.html and test thoroughly making sure your code does not hit any errors. Show your lecturer your code for verification.

# Complete the problem statement

1. In searchMovie.js, create a click callback function on selector btnSearch.
   * Retrieve title, year and plot from the form fields
   * Make ajax call to OMDB API (<http://www.omdbapi.com>) using GET method with parameters t=title, y=year and plot=plot (See <http://www.omdbapi.com/#parameters>)
   * **Important:** When making ajax call to OMDB API, you need to get the API key from <http://www.omdbapi.com/apikey.aspx>, also you need to set the parameter dataType: "jsonp"



* + Upon successful response,
    - retrieve the Title, Released, Runtime, Genre, Actors and Plot attributes from the response object and append them to the selector contents. Retrieve the Poster attribute from the response object to display as an image using the selector poster.
    - Add the response object into an array to be stored in local storage (Similar to Step 21)

1. Run searchMovie.php and enter a movie title and year and select either short/full plot and click on the Search button. If coded correctly, the details should be displayed below the form on the left column and the movie image on the right column.
2. In index.js, retrieve the reviews from localStorage and display each movie details in a Bootstrap panel with the movie title in the panel heading and the movie plot in the panel body.