PERSISTENT STORAGE

***What is LOCAL-STORAGE?***

**Local storage** is more secure, and large amounts of data can be **stored locally**, without affecting website performance. Unlike cookies, the **storage** limit is far larger and information is never transferred to the server.

***What is HTML Local-Storage?***

With local storage, web applications can store data locally within the user's browser. Before HTML5, application data had to be stored in cookies, included in every server request. Local storage is more secure, and large amounts of data can be stored locally, without affecting website performance. Unlike cookies, the storage limit is far larger and information is never transferred to the server. Local storage is per origin. All pages, from one origin, can store and access the same data.

***Why we use HTML LOCAL STORAGE?***

Storing information locally on a user’s computer is a powerful strategy for a developer who is creating something for the Web.

***BENEFITS OF LOCAL STORAGE:***

This feature can be implemented using the combination of JavaScript and HTML 5.

You should differentiate this feature from that of using cookies. Below are some of the advantages that HTML 5 storage has when compared to cookie-based storage:

1. The data is stored totally on the client and does not shuttle between the client and server on each request or response.
2. The size of the storage space is large when compared to that of a cookie.
3. Session-based storage is available.

***IMPLEMENTATION OF LOCAL STORAGE:***

HTML local storage provides two objects for storing data on the client:

* **window.localStorage** - stores data with no expiration date
* **window.sessionStorage** - stores data for one session (data is lost when the browser tab is closed) before using local storage, check browser support for local Storage and session Storage.

***if (typeof(Storage) !== "undefined") {  
    // Code for localStorage/sessionStorage.  
} else {  
    // Sorry! No Web Storage support...  
}***

The localStorage Object

The localStorage object stores the data with no expiration date. The data will not be deleted when the browser is closed, and will be available the next day, week, or year.

**//Store  
localStorage.setItem("lastname", "Smith");  
//Retrieve  
document.getElementById("result").innerHTML = localStorage.getItem("lastname");**

**FOR EXAMPLE:**

***<!DOCTYPE html>***

***<html>***

***<head>***

***<script>***

***function clickCounter() {***

***if(typeof(Storage) !== "undefined") {***

***if (localStorage.clickcount) {***

***localStorage.clickcount = Number(localStorage.clickcount)+1;***

***} else {***

***localStorage.clickcount = 1;***

***}***

***document.getElementById("result").innerHTML = "You have clicked the button " + localStorage.clickcount + " time(s).";***

***} else {***

***document.getElementById("result").innerHTML = "Sorry, your browser does not support web storage...";***

***}***

***}***

***</script>***

***</head>***

***<body>***

***<p><button onclick="clickCounter()" type="button">Click me!</button></p>***

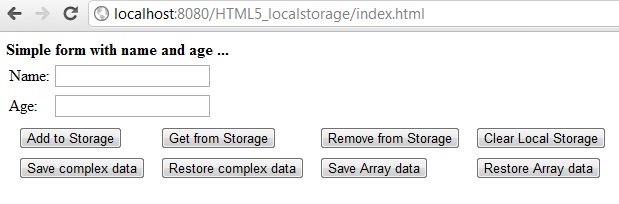
***<div id="result"></div>***

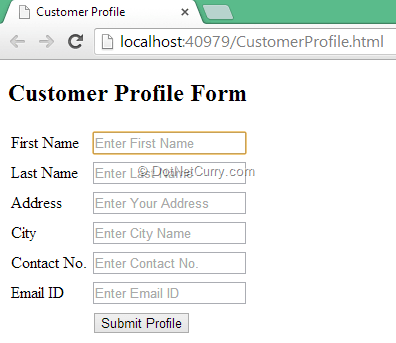
***<p>Click the button to see the counter increase.</p>***

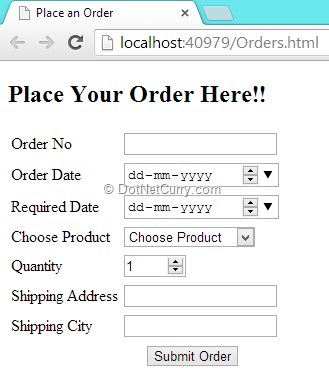
***<p>Close the browser tab (or window), and try again, and the counter will continue to count (is not reset).</p>***

***</body>***

***</html>***







***BROWSER SUPPORT:***

The support for web storage is pretty good for an HTML5 specification; it is supported by all the major browsers and even IE8, so the only thing you might need to be wary of is IE7 if you’re still supporting that.

***CONCLUSION:***

Local storage in small apps like this can very welcome substitutes for databases. Storing small amounts of data doesn’t need to be complex.

