

## **Local storage with HTML 5**

- With web 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. Web storage is more secure, and large amounts of data can be stored locally, without affecting website performance.

HTML web 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)

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

- Create a localStorage name/value pair with name="lastname" and value="Smith"
- Retrieve the value of "lastname" and insert it into the element with id="result"

The example above could also be written like this:

- `// Store`  
`localStorage.lastname = "Smith";`  
`// Retrieve`  
`document.getElementById("result").innerHTML = localStorage.lastname;`

- The syntax for removing the "lastname" localStorage item is as follows:
- `localStorage.removeItem("lastname");`

## Example for Session Storage

The following example counts the number of times a user has clicked a button, in the current session:

Example

- ```
if (sessionStorage.clickcount) {  
    sessionStorage.clickcount = Number(sessionStorage.clickcount) + 1;  
} else {  
    sessionStorage.clickcount = 1;  
}  
document.getElementById("result").innerHTML = "You have clicked  
the button " +  
sessionStorage.clickcount + " time(s) in this session.";
```



Click Me

- Click the button to see the counter increase.
- Close the browser tab (or window), and try again, and the counter is reset.