

HTML 5

Web Storage

HTML



Bok, Jong Soon
jongsoon.bok@gmail.com
www.javaexpert.co.kr

What is Web Storage ?

- In HTML5, web pages can store data locally within the user's browser.
- Earlier, this was done with cookies.
- However, Web Storage is more secure and faster.
- The data is not included with every server request, but used ONLY when asked for.



What is Web Storage ?

- It is also possible to store large amounts of data, without affecting the website's performance.
- The data is stored in key/value pairs, and a web page can only access data stored by itself.

Cookies

- Are overly complicated.
 - `document.cookie = "foo=bar; path=/";`
- Is a session-based cookie.
- If I want to store something for longer...
 - `document.cookie = "foo=bar; path=/; expires=Tues, 13 Sept 2014 12:00:00";`
- To delete a cookie, I need to set the value to blank:
 - `document.cookie = "foo=; path=/";`
- Typically has a limit of 4KB.

Cookies vs. Web Storage

- With Cookies, IE6 supports only 20 cookies per domain and a maximum size of 4KB per cookie.
- Web Storage doesn't have a limit per domain and limits per domain are upwards of 5MB.

Cookies vs. Web Storage (Cont.)

- With Cookies, IE6 supports only 20 cookies per domain and a maximum size of 4KB per cookie.
- Web Storage doesn't have a limit per domain and limits per domain are upwards of 5MB.

Lab1 : Check Web Browsers for localStorage and sessionStorage

■ Web Browsers

- IE10, Firefox, Google Chrome, Opera, Safari

■ Text Editors

- Notepad++ or Editplus

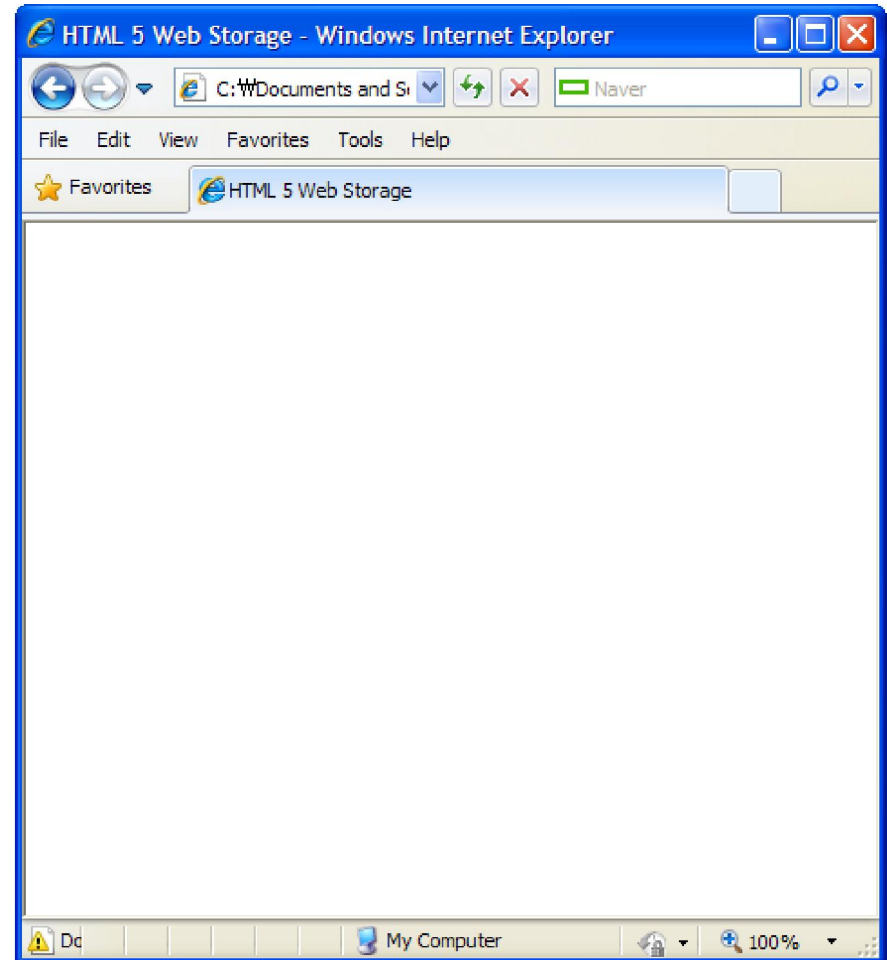
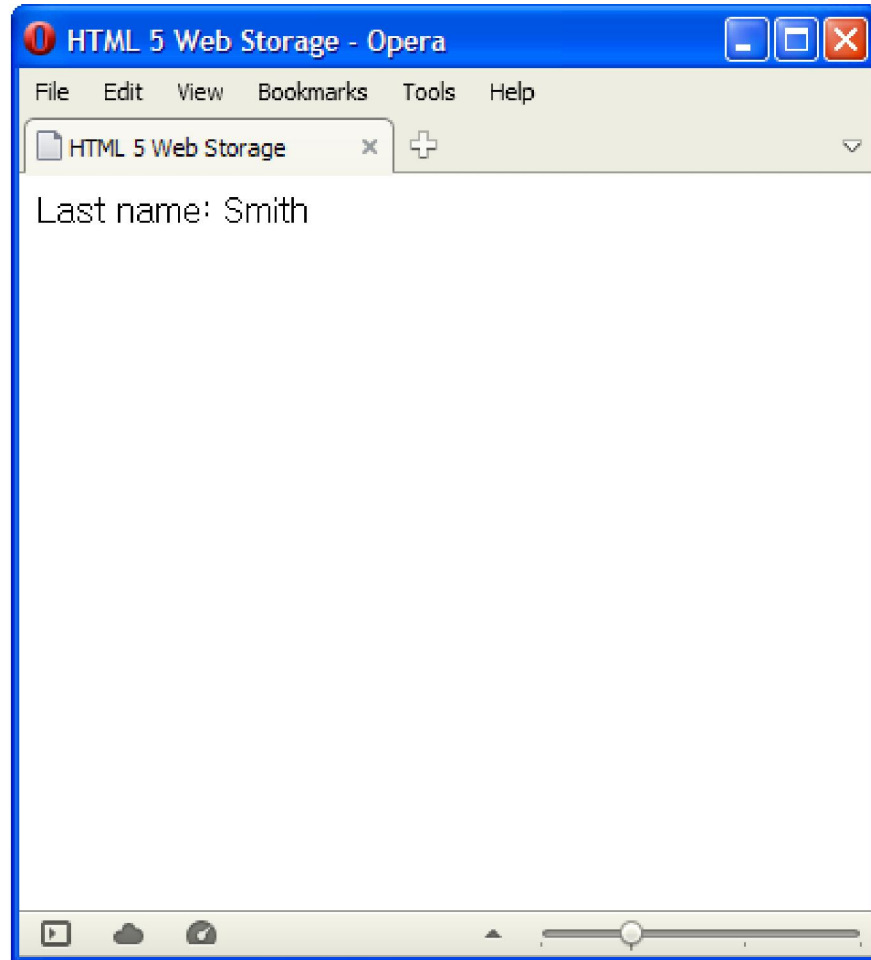
■ Files

- `checkstorage.html`

Lab1 : checkstorage.html

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title> HTML 5 Web Storage </title>
5     <meta charset="utf-8">
6     <script type="text/javascript">
7       function getData(){
8         if(typeof(Storage)!="undefined") {
9           localStorage.lastname="Smith";
10          document.getElementById("result").innerHTML=
11            "Last name: " + localStorage.lastname;
12        } else {
13          document.getElementById("result").innerHTML=
14            "Sorry, your browser does not support web storage...";
15        }
16      }
17    </script>
18  </head>
19  <body onload="getData()">
20    <div id="result"></div>
21  </body>
22 </html>
```


Lab1 : Result



localStorage and sessionStorage

- There are two new objects for storing data on the client:
 - **localStorage** - stores data with no expiration date
 - **sessionStorage** - stores data for one session

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.
- Is based around the domain and spans all windows that are open on that domain.
- If you set some data on local storage it immediately becomes available on any other window on the same domain.

The `localStorage` Object (Cont.)

- You can close your browser, reboot your machine, come back to it days later, and the data will still be there.

Lab2 : localStorage

■ Web Browsers

- IE10, Firefox, Google Chrome, Opera, Safari

■ Text Editors

- Notepad++ or Editplus

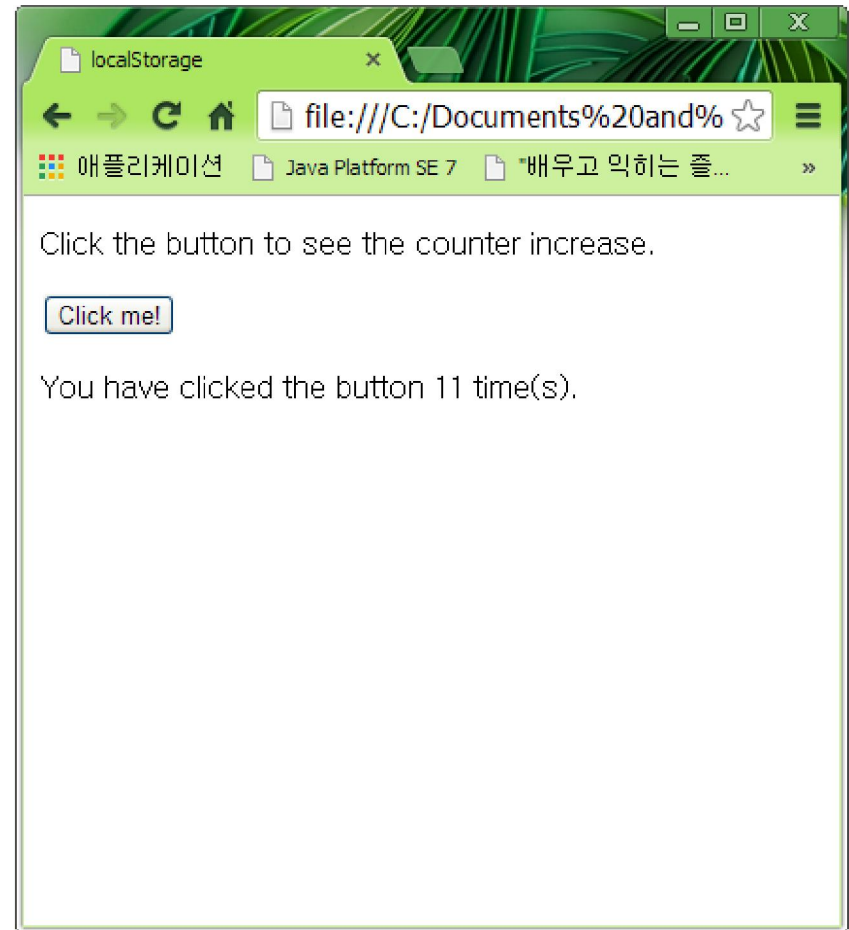
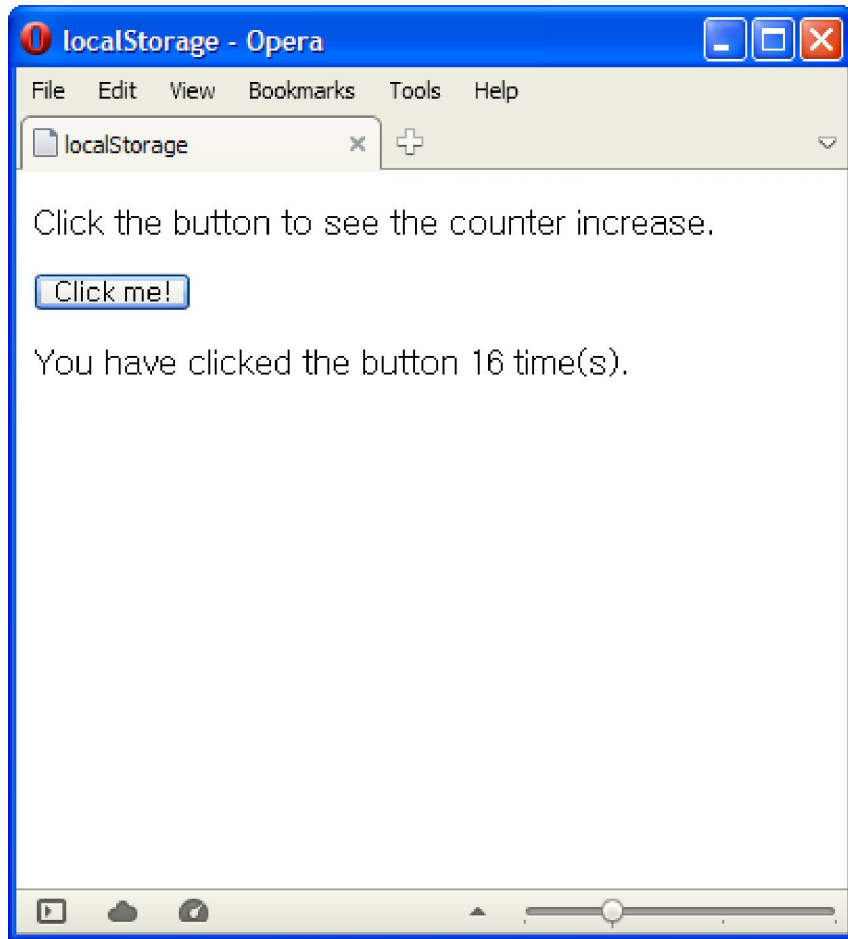
■ Files

- localStorage.html

La

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>localStorage </title>
5     <script type="text/javascript">
6       function clickCounter() {
7         if(typeof(Storage)!="undefined") {
8           if (localStorage.clickcount) {
9             localStorage.clickcount=Number(localStorage.clickcount)+1;
10          } else {
11            localStorage.clickcount=1;
12          }
13          document.getElementById("result").innerHTML=
14            "You have clicked the button " +
15            localStorage.clickcount + " time(s).";
16        } else {
17          document.getElementById("result").innerHTML=
18            "Sorry, your browser does not support web storage...";
19        }
20      }
21    </script>
22  </head>
23  <body>
24    <p>Click the button to see the counter increase.</p>
25    <p>
26      <button onclick="clickCounter()" type="button">Click me!</button></p>
27    <div id="result"></div>
28  </body>
29 </html>
```

Lab2 : Result



The `sessionStorage` Object

- Is equal to the `localStorage` object, **except** that it stores the data for only one session.
- The data is deleted when the user closes the browser window.
- If you opened another window on the same domain, it wouldn't have access to that session data.

Lab3 : sessionStorage

- Web Browsers

- IE10, Firefox, Google Chrome, Opera, Safari

- Text Editors

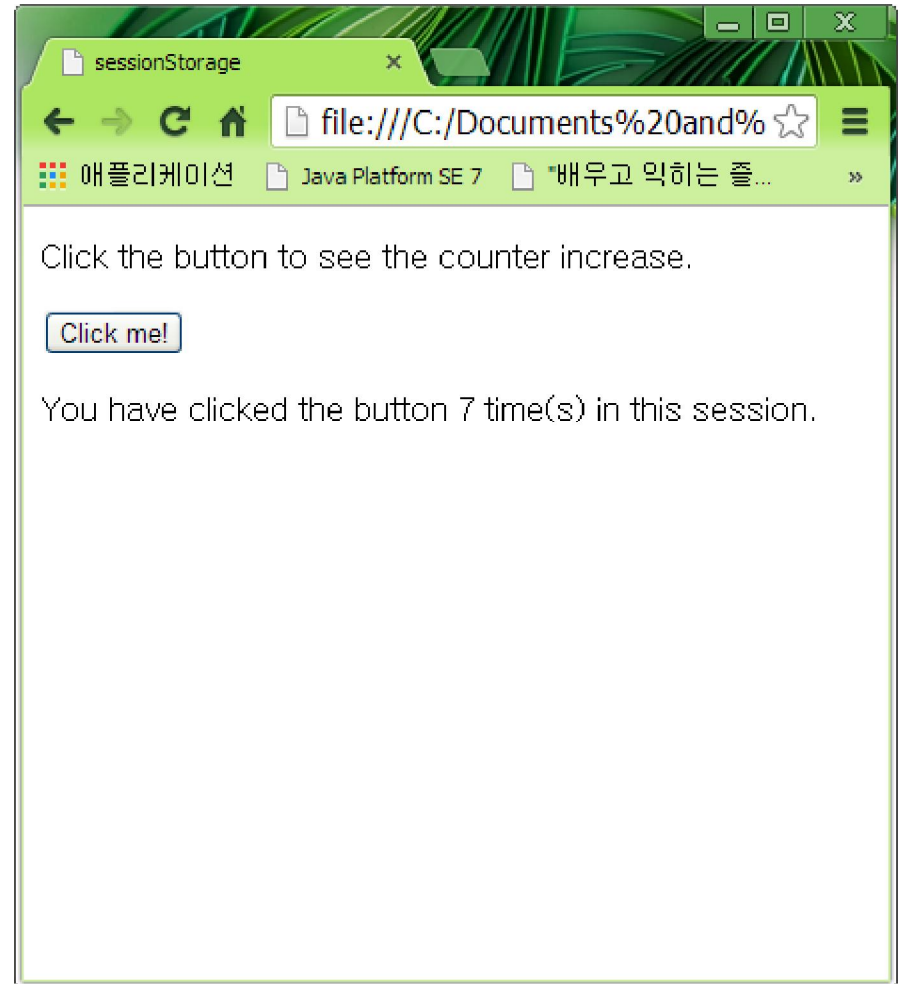
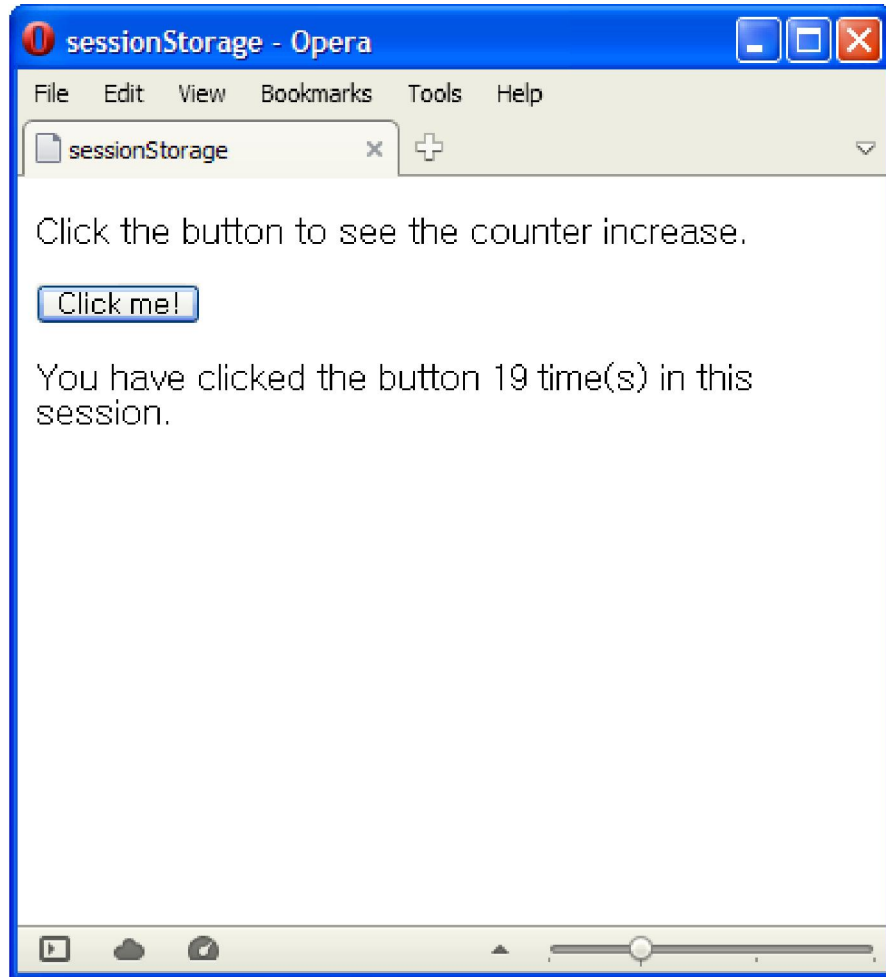
- Notepad++ or Editplus

- Files

- `sessionstorage.html`

```
<head>
  <title>sessionStorage </title>
  <script>
    function clickCounter() {
      if(typeof(Storage)!="undefined") {
        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.";
      } else {
        document.getElementById("result").innerHTML=
          "Sorry, your browser does not support web storage...";
      }
    }
  </script>
</head>
<body>
  <p>Click the button to see the counter increase.</p>
  <p><button onclick="clickCounter()" type="button">Click me!</button></p>
  <div id="result"></div>
```

Lab3 : Result



An Overview of the API

- readonly attribute unsigned long length
- getter DOMString key(in unsigned long index)
- getter any getItem(in DOMString key);
- setter creator void setItem(
 in DOMString key, in any data);
- deleter void removeItem(in DOMString key);
- void clear();

An Overview of the API (Cont.)

```
sessionStorage.setItem('twitter', '@rem');  
alert( sessionStorage.getItem('twitter') ); //  
    shows @rem
```

```
sessionStorage.setItem('cost', 12);  
alert(sessionStorage.getItem('cost'));  
// shows 12
```

An Overview of the API (Cont.)

```
for (var i=0;i<sessionStorage.length;i++) {  
    alert( sessionStorage.key(i) + '=' +  
    sessionStorage.getItem(  
        sessionStorage.key(i) ) );  
}
```

An Overview of the API (Cont.)

```
sessionStorage.setItem( 'bruce' ,  
    "Think he's a professor of synergies" );  
sessionStorage.setItem( 'remy' ,  
    "Think Bruce is a loony professor  
    of synergies" );  
alert( sessionStorage.length ); // shows 2  
sessionStorage.removeItem( 'bruce' );  
alert( sessionStorage.length ); // show 1  
sessionStorage.clear();  
alert( sessionStorage.length ); // shows 0
```

WebKit's Developer Tools

The screenshot shows the Mozilla Firefox browser window with the WebKit Developer Tools open. The browser's address bar shows 'localStorage'. The page content includes a button labeled 'Click me!' and a message 'You have clicked the button 1 time(s)'. The Developer Tools panel is open to the 'DOM' tab, displaying a tree view of the document's structure. A red rectangle highlights the 'localStorage' object, which contains a single item named 'clickcount' with the value '1'.

localStorage - Mozilla Firefox

File Edit View History Bookmarks Tools Help

localStorage +

Click the button to see the counter increase.

Click me!

You have clicked the button 1 time(s).

Console HTML CSS Script DOM Net Cookies

window

- history 2 history entries
- indexedDB IDBFactory { open=open(), deleteDatabase=deleteDatabase(), cmp=cmp() }
- innerHeight 150
- innerWidth 849
- length 0
- localStorage 1 item in Storage clickcount="1"**
 - clickcount "1"**
- location file:///C:/Documents%20and%20Settings/Redmond/My%20Documents/Text/HTML5/Chapter%2015/localstorage.html { constructor={...}, href="file:///C:/Documents%20...%2015/localstorage.html", pathname="/C:/Documents%20and%20S...%2015/localstorage.html", more... }
- locationbar BarProp { visible=true }
- menubar BarProp { visible=true }
- mozAnimationStartTime 1382885566134
- mozIndexedDB IDBFactory { open=open(), deleteDatabase=deleteDatabase(), cmp=cmp() }
- mozInnerScreenX 100

WebKit's Developer Tools

