<https://www.cnblogs.com/yuzhongwusan/archive/2011/12/19/2293347.html>

html5中的Web Storage包括了两种存储方式：sessionStorage和localStorage。

**sessionStorage**用于本地存储一个会话（session）中的数据，这些数据只有在同一个会话中的页面才能访问并且当会话结束后数据也随之销毁。因此sessionStorage不是一种持久化的本地存储，仅仅是会话级别的存储。

而**localStorage**用于持久化的本地存储，除非主动删除数据，否则数据是永远不会过期的。

## web storage和cookie的区别

Web Storage的概念和cookie相似，区别是它是为了更大容量存储设计的。Cookie的大小是受限的，并且每次你请求一个新的页面的时候Cookie都会被发送过去，这样无形中浪费了带宽，另外cookie还需要指定作用域，不可以跨域调用。

除此之外，**Web Storage**拥有setItem,getItem,removeItem,clear等方法，不像cookie需要前端开发者自己封装setCookie，getCookie。

但是Cookie也是不可以或缺的：Cookie的作用是与服务器进行交互，作为HTTP规范的一部分而存在 ，而Web Storage仅仅是为了在本地“存储”数据而生（来自@otakustay 的纠正）

## html5 web storage的浏览器支持情况

浏览器的支持除了IE７及以下不支持外，其他标准浏览器都完全支持(ie及FF需在web服务器里运行)，值得一提的是IE总是办好事，例如IE7、IE6中的UserData其实就是**javascript本地存储**的解决方案。通过简单的代码封装可以统一到所有的浏览器都支持web storage。

要判断浏览器是否支持localStorage可以使用下面的代码：

**if**(window.localStorage){ alert("浏览支持localStorage") }**else**{ alert("浏览暂不支持localStorage") } *//或者* **if**(**typeof** window.localStorage == 'undefined'){ alert("浏览暂不支持localStorage") }

## localStorage和sessionStorage操作

localStorage和sessionStorage都具有相同的操作方法，例如setItem、getItem和removeItem等

### localStorage和sessionStorage的方法

#### setItem存储value

用途：将value存储到key字段  
用法：.setItem( key, value)  
代码示例：

sessionStorage.setItem("key", "value"); localStorage.setItem("site", "js8.in");

#### getItem获取value

用途：获取指定key本地存储的值  
用法：.getItem(key)  
代码示例：

**var** value = sessionStorage.getItem("key"); **var** site = localStorage.getItem("site");

#### removeItem删除key

用途：删除指定key本地存储的值  
用法：.removeItem(key)  
代码示例：

sessionStorage.removeItem("key"); localStorage.removeItem("site");

#### clear清除所有的key/value

用途：清除所有的key/value  
用法：.clear()  
代码示例：

sessionStorage.clear(); localStorage.clear();

#### 其他操作方法：点操作和[]

web Storage不但可以用自身的setItem,getItem等方便存取，也可以像普通对象一样用点(.)操作符，及[]的方式进行数据存储，像如下的代码：

**var** storage = window.localStorage; storage.key1 = "hello"; storage["key2"] = "world"; console.log(storage.key1); console.log(storage["key2"]);

#### localStorage和sessionStorage的key和length属性实现遍历

sessionStorage和localStorage提供的key()和length可以方便的实现存储的数据遍历，例如下面的代码：

**var** storage = window.localStorage; **for** (**var** i=0, len = storage.length; i < len; i++){ **var** key = storage.key(i); **var** value = storage.getItem(key); console.log(key + "=" + value); }

### storage事件

storage还提供了**storage事件**，当键值改变或者clear的时候，就可以触发storage事件，如下面的代码就添加了一个storage事件改变的监听：

**if**(window.addEventListener){ window.addEventListener("storage",handle\_storage,**false**); }**else** **if**(window.attachEvent){ window.attachEvent("onstorage",handle\_storage); } **function** handle\_storage(e){ **if**(!e){e=window.event;} }

storage事件对象的具体属性如下表：

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| key | String | The named key that was added, removed, or moddified |
| oldValue | Any | The previous value(now overwritten), or null if a new item was added |
| newValue | Any | The new value, or null if an item was added |
| url/uri | String | The page that called the method that triggered this change |

## Web Storage Demo

[HTML5 Demos: Storage](http://html5demos.com/storage)     [Web Storage Example](http://people.opera.com/shwetankd/external/demos/webstorage_demo.htm)

标签: [yuzhongwusan](http://www.cnblogs.com/yuzhongwusan/tag/yuzhongwusan/), [HTML5](http://www.cnblogs.com/yuzhongwusan/tag/HTML5/), [前端开发](http://www.cnblogs.com/yuzhongwusan/tag/%E5%89%8D%E7%AB%AF%E5%BC%80%E5%8F%91/)