Web storage

window.localStorage和window.sessionStorage. 这两个对象均是Storage类的两个实例，自然也具有Storage类的属性和方法。

Storage 属性方法的说明：

length 获取存储数据的条数

key(n)　　　　 n:索引值 根据索引值，返回键名

getItem(key) key:键名 根据键名，获取数据值

setItem(key,value) key:键名 value:键值 根据键名和键值设置数据项，如果键名已经存在，则覆盖值

removeItem(key) key:键名 根据键名删除一个数据项

clear() 清空当前的Storage对象

可以方便的在web请求之间保存数据。有了本地数据，就可以避免数据在浏览器和服务器间不必要地来回传递

sessionStorage、localStorage、cookie都是在浏览器端存储的数据，

其中sessionStorage的概念很特别，引入了一个“浏览器窗口”的概念。

sessionStorage是在同源的同窗口（或tab）中，始终存在的数据。也就是说只

要这个浏览器窗口没有关闭，即使刷新页面或进入同源另一页面，数据仍然存在。关闭窗口后，sessionStorage即被销毁。

同时“独立”打开的不同窗口，即使是同一页面，sessionStorage对象也是不同的。

Web Storage带来的好处：

减少网络流量：一旦数据保存在本地后，就可以避免再向服务器请求数据，因此减少不必要的数据请求，减少数据在浏览器和服务器间不必要地来回传递。

快速显示数据：性能好，从本地读数据比通过网络从服务器获得数据快得多，本地数据可以即时获得。再加上网页本身也可以有缓存，因此整个页面和数据都在本地的话，可以立即显示。

临时存储：很多时候数据只需要在用户浏览一组页面期间使用，关闭窗口后数据就可以丢弃了，这种情况使用sessionStorage非常方便

sessionStorage 、localStorage 和 cookie 之间的区别

共同点：都是保存在浏览器端，且同源(协议，主机名，端口一致)的。

区别：cookie数据始终在同源的http请求中携带（即使不需要），即cookie在浏览器和服务器间来回传递。

而sessionStorage和localStorage不会自动把数据发给服务器，仅在本地保存。cookie数据还有路径（path）的概念，可以限制cookie只属于某个路径下。

存储大小限制也不同:

cookie数据不能超过3-4k（4 kb =4096b），同时因为每次http请求都会携带cookie，所以cookie只适合保存很小的数据，如会话标识。

sessionStorage和localStorage 虽然也有存储大小的限制，但比cookie大得多，可以达到5M或更大。

sessionStorage 在同一个浏览器窗口或者是同一个页面中。

localStorage 在所有同源窗口中都是共享的；

cookie也是在所有同源窗口中都是共享的。

localStorage（本地存储）,可以长期存储数据，没有时间限制，除非手动删除。

sessionStorage（会话存储），标签页或者浏览器关闭，存储数据就会消失。

cookie只在设置的cookie过期时间之前一直有效，即使窗口或浏览器关闭。

习惯用法：

sessionStorage 用在服务端比较多，如果web服务器端使用的是session，那么所有的数据都保存在服务器上，客户端每次请求服务器的时候会发送当前会话的sessionid，服务器根据当前sessionid判断相应的用户数据标志，以确定用户是否登录或具有某种权限。由于数据是存储在服务器上面，所以你不能伪造，但是如果你能够获取某个登录用户的 sessionid，用特殊的浏览器伪造该用户的请求也是能够成功的。sessionid是服务器和客户端链接时候随机分配的，一般来说是不会有重复，但如果有大量的并发请求，也不是没有重复的可能性。

如果浏览器使用的是cookie，那么所有的数据都保存在浏览器端，比如你登录以后，服务器设置了cookie用户名，那么当你再次请求服务器的时候，浏览器会将用户名一块发送给服务器，这些变量有一定的特殊标记。服务器会解释为cookie变量，所以只要不关闭浏览器，那么cookie变量一直是有效的，所以能够保证长时间不掉线。如果你能够截获某个用户的 cookie变量，然后伪造一个数据包发送过去，那么服务器还是认为你是合法的。所以，使用 cookie被攻击的可能性比较大。如果设置了的有效时间，那么它会将 cookie保存在客户端的硬盘上，下次再访问该网站的时候，浏览器先检查有没有 cookie，如果有的话，就读取该 cookie，然后发送给服务器。如果你在机器上面保存了某个论坛 cookie，有效期是一年，如果有人入侵你的机器，将你的 cookie拷走，然后放在他的浏览器的目录下面，那么他登录该网站的时候就是用你的的身份登录的。所以 cookie是可以伪造的。当然，伪造的时候需要主意，直接copy cookie文件到 cookie目录，浏览器是不认的，他有一个index.dat文件，存储了 cookie文件的建立时间，以及是否有修改，所以你必须先要有该网站的 cookie文件，并且要从保证时间上骗过浏览器。

为了安全着想，将一些重要的信息，例如判断是否登录，或者是否具有某种权限等信息，存放在服务器上的session。其他信息可以放在客户的浏览器上的cookie。

## 兼容问题

Chrome4+ Firefox3.5+ Firefox1.5+ IE8+ Safari 4+ Opera10.5+