## localstorage 必知必会

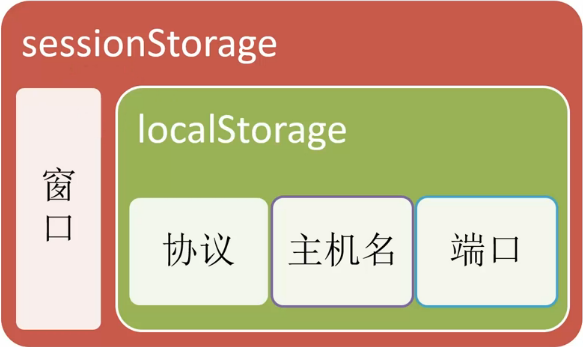
HTML5的storage是存储在你的计算机上，网站在页面加载完毕后可以通过Javascript来获取这些数据。在HTML5中，本地存储是一个window的属性，包括localStorage和sessionStorage，前者是一直存在本地的，后者只是伴随着session，窗口一旦关闭就没了。二者用法完全相同，

<https://www.runoob.com/w3cnote/localstorage-spec.html>

localstorage 在浏览器的 API 有两个：localStorage 和sessionStorage，存在于 window 对象中：localStorage 对应 window.localStorage，sessionStorage 对应 window.sessionStorage。

localStorage 和 sessionStorage 的区别主要是在于其生存期。

**基本使用方法**



* 这里的作用域指的是：如何隔离开不同页面之间的localStorage（总不能在百度的页面上能读到腾讯的localStorage吧，哈哈哈）。
* localStorage只要在相同的协议、相同的主机名、相同的端口下，就能读取/修改到同一份localStorage数据。
* sessionStorage比localStorage更严苛一点，除了协议、主机名、端口外，还要求在同一**窗口**（也就是浏览器的标签页）下。

## 生存期

localStorage理论上来说是永久有效的，即不主动清空的话就不会消失，即使保存的数据超出了浏览器所规定的大小，也不会把旧数据清空而只会报错。但需要注意的是，在移动设备上的浏览器或各Native App用到的WebView里，localStorage都是不可靠的，可能会因为各种原因（比如说退出App、网络切换、内存不足等原因）被清空。  
sessionStorage的生存期顾名思义，类似于session，只要关闭浏览器（也包括浏览器的标签页），就会被清空。由于sessionStorage的生存期太短，因此应用场景很有限，但从另一方面来看，不容易出现异常情况，比较可靠。

## 数据结构

localstorage为标准的键值对（Key-Value,简称KV）数据类型，简单但也易扩展，只要以某种编码方式把想要存储进localstorage的对象给转化成字符串，就能轻松支持。举点例子：把对象转换成json字符串，就能让存储对象了；把图片转换成DataUrl（base64），就可以存储图片了。另外对于键值对数据类型来说，"键是唯一的"这个特性也是相当重要的，重复以同一个键来赋值的话，会覆盖上次的值。

**过期时间**

很遗憾，localstorage原生是不支持设置过期时间的，想要设置的话，就只能自己来封装一层逻辑来实现：