Local Storage和Session Storage区别在与存储期和作用域。

“Web存储”草案指出，这两个对象应该可以存储结构化数据（数组和对象），但实际上浏览器目前还是只支持存储字符串的。要想存储其他数据，需要进行编码和解码。例如存储照片，利用canvas。

Local Storage：

1. 作用域  
   限制在文档源，同源的文档共享同样的Local Storage。Local Storage的作用域还受到浏览器厂商的限制，不同的浏览器访问的Local Storage也不同。
2. 有效期  
   永久性，除非手动删除。

Session Storage：

1. 有效期  
   与存储数据的脚本（H5存储当然只能有客户端JS设置，服务端无从下手，甚至你不发送，服务器端都不知道你存储什么）所在的最顶层窗口或者是浏览器标签页是一样的。一旦窗口或者浏览器关闭了，session storage就删除了。Session本身的意思就是会话，它只为维持会话而生。
2. 作用域  
   同源。还被限定在窗口中。是窗口，而不是iframe。即使同源的不同标签中，他们各自拥有不同的session storage，，无法相互读取和覆盖，甚至是同一个页面和脚本在不同的标签页中都无法覆盖和读取。但是在同一个标签页的不同的iframe是可以相互读取的。

各种API？

1. getItem（）
2. setItem（）
3. removeItem（）
4. clear（）
5. length
6. key（），传入0~length-1的数字

存储事件？

当Local Storage发生改变的时候，浏览器会在其他对该数据可见的窗口对象上（window 对象）触发存储事件，但是，对数据改变的窗口对象上自身是不会触发的。

window.onstorage，其事件对象有以下属性：

1. key 被设置或改变或移除的值，如果是clear（），为null
2. oldValue
3. newValue
4. url 触发该存储改变的脚本所在的文档的URL
5. storageArea 说明改变的是local还是session