

web存储

1. 本地存储

- a. 临时存储: `sessionStorage`
- b. 永久存储: `localStorage`

2. cookie

一、localStorage:

作用: 在HTML5中, 新加入了一个localStorage特性, 这个特性主要是 用来作为本地存储来使用的, 解决了cookie存储空间不足的问题 (cookie中每条cookie的存储空间为4k), localStorage中一般浏览器支持的是5M大小, 这个在不同的浏览器中localStorage会有 所不同。

语法: (注意 localStorage 只支持 string 类型的数据存储)

```
1 // 存数据
2 localStorage.setItem('key', 'value')
3
4 // 读取数据
5 localStorage.getItem('key')
6
7 // 删除一条数据
8 localStorage.removeItem('key')
9
10 // 清除所有数据
11 localStorage.clear()
```

二、sessionStorage:

作用: sessionStorage 用于临时保存同一窗口(或标签页)的数据, 在关闭窗口或标签页之后将会删除这些数据。

语法: 和localStorage一样

三、cookie:

作用: 浏览器存储数据的方式 浏览器发起请求的时候, 会自动把cookie中存储的内容, 携带并发送到服务器, **容量: 4k**。

获取: `document.cookie`

添加cookie: `document.cookie = "key=value"`

所有的cookie都是单会话cookie, 即浏览器关闭后这些cookie将会丢失, 事实上这些cookie仅仅是存储在内存中, 而没有建立相应的硬盘文件。在实际开发中, cookie常常需要**长期**保存, 例如保存用户登录的状态。这可以用下面的选项来实现:

`document.cookie="userId=828;expires=GMT_String";`

其中GMT_String是以GMT格式表示的时间字符串, 这条语句就是将userId这个cookie设置为GMT_String表示的过期时间, 超过这个时间, cookie将消失, 不可访问。如果要将cookie设置为10天后过期, 可以这样实现:

```
1 // GMT_String: new Date().toGMTString();
2 var date=new Date();
3 var expiresDays=10; //将date设置为10天以后的时间
4 date.setTime(date.getTime()+expiresDays*24*3600*1000); //将userId和userName两个cookie设置为10天后过期
5 document.cookie="userId=828; userName=hulk;expires="+date.toGMTString();
```