本地数据存储

cookie

如何存储

http传输

存储个数

存储大小

存储时间

会话结束

设置结束时间：expires 和 max-age

访问限制

子域名限制

路径限制

http only

localStorage、sessionStorage 和 cookie的区别

如何存储：相同

http传输：不同

存储个数：不同

存储大小：不同

存储时间：不同

访问限制：不同

localStorage 和 sessionStorage 的区别

存储时间：不同

访问限制：不同

localStorage、sessionStorage的优势 和 局限

优势：存储空间变大；节约带宽

局限：

高级浏览器

需要转换类型

读取

页面变卡

不利于爬虫

localStorage 的理解

sessionStorage 的理解

cookie

定义：

服务器与客户端进行数据传输的一种方式，cookie的信息是通过头信息发送的，浏览器接收到cookie以后会自动的保存在本地，并且是按照域名进行分类存储的，浏览器会在每次请求的时候把与该次请求同域下的cookie自动通过请求头信息发。

特性：

**1. 如何存储**：cookie是按照域名进行存储发送

**2. http传输**： 客户端请求时，浏览器会自动将该次请求同域下的cookie自动放入头信息传给服务端

**3. 存储个数**：每个域名下可以存储多个cookie，不同的浏览器对cookie的个数有一定的限制，大概100左右，多个cookie通过 分号+空格 进行分隔

**4. 存储大小**：每一个cookie的内容大小也是有限制的，不同浏览器的单个cookie的内容限制也不一样，大概4000字节，单个cookie通过 等号 来分隔 key与value

**5. 存储时间**：

**会话结束**：cookie的存储也是有时间上的限制，默认是浏览器会话结束的时候，即关闭浏览器的时候；

**设置结束时间**：存储某个cookie的时候给他设置过期的时间，设置的方式有两种

- **expires**：早期使用的，现在不推荐，设置是到期的时间。比如服务器设置年月日，客户端时间和服务器这个设置时间对比，时间到cookie就删除。存在问题：比如服务器设置的时间比客户端当前时间还要早，cookie就设置不上去了。

- **max-age**：最新的方式，推荐使用，设置保存时间，毫秒单位。比如设置保留2天，客户端会根据当前存储时间为基准，到2天再删除cookie。

**6. 访问限制**：访问cookie还受到子域名与路径的影响

**子域名限制**：不同子域的cookie也是有访问限制的

- baidu.com 下的cookie与study.baidu.com 下的cookie是分开存储的，子域可以访问父域下的cookie，但是父域不能访问子域的内容

**路径限制**：不同的路径存放的cookie也是有访问限制的

- baidu.com/a 下的cookie与 baidu.com/b 的cookie也是分开的，子路径下能访问父路径的cookie，父路径不能访问子路径的cookie送给服务器。

**7. http only**：

当某个cookie设置了http only以后，那么这个cookie就只能使用在http协议传输上，其他方式都不能去操作他

localStorage、sessionStorage

定义：

客户端在操作页面的时候，页面设置了对localStorage、sessionStorage进行设置、修改、删除，并且将结果按照域名进行分类存储在本地， 服务器与客户端进行数据传输的一种方式，cookie的信息是通过头信息发送的，浏览器接收到cookie以后会自动的保存在本地，并且是按照域名进行分类存储的，浏览器会在每次请求的时候把与该次请求同域下的cookie自动通过请求头信息发。

特性：

**1. 如何存储**：是按照域名进行存储发送

**2. http传输**：不会随着http请求一起发送给服务器

**3. 存储个数**：没有限制

**4. 存储大小**：存储的数据大概5M

**5. 存储时间**：

localStorage：没有过期时间，永久存储的，除非手动删除

sessionStorage：浏览器会话结束的时候自动销毁

**6. 访问限制**：

localStorage：同域共享，其中有一个页面更新，其他页面通过storage事件会自动更新。

sessionStorage：只能在当前设置的页面进行访问。

localStorage、sessionStorage的**优势 和 局限**

**优势 ：**

1、存储空间变大

2、节约带宽，但是这个却是只有在高版本的浏览器中才支持的

**局限：**

1、**高级浏览器**：浏览器的大小不统一，并且在IE8以上的IE版本才支持localStorage这个属性

2、**需要转换类型**：目前所有的浏览器中都会把localStorage、sessionStorage的值类型限定为string类型，这个在对我们日常比较常见的JSON对象类型需要一些转换

3、读取：localStorage、sessionStorage在浏览器的隐私模式下面是不可读取的=

4、**页面变卡**：localStorage、sessionStorage本质上是对字符串的读取，如果存储内容多的话会消耗内存空间，会导致页面变卡

5、**不利于爬虫**：localStorage、sessionStorage不能被爬虫抓取到

cookie 的理解

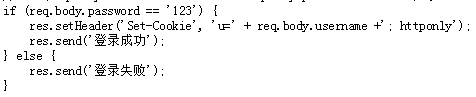
cookie是服务端和客户端的数据传输的方式。

两次http请求是独立的，这个是http协议的特性，工作完成之后断开连接。这样有利于提高性能，节约服务器资源。所以需要有一个中间体来联系两次请求状态。

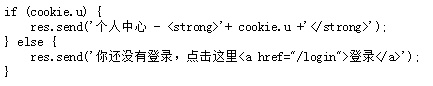
服务端如何判断客户端是**否是登录状态**：当用户登录成功以后，为了能够让服务器识别每次请求的客户端的登录信息，服务器会在每次登录验证成功以后，发送一个登录的凭证给客户端，客户端接收到凭证以后，保存在自己的客户端机器上，下面访问该服务器的时候把这个凭证给带过来，那么服务器就可以根据这个凭证判断当前请求的客户端是否是登录的状态——这个凭证其实就是 cookie。

传输：cookie数据是包装在头信息进行发送的：Set-Cookie:数据

服务端发送凭证：



服务端接收凭证：



客户端也可以设置cookie

设置：**document**.**cookie** = **'a=1; path=/'**;

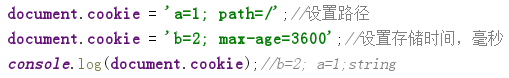
获取：获取到的是字符串**document**.**cookie**

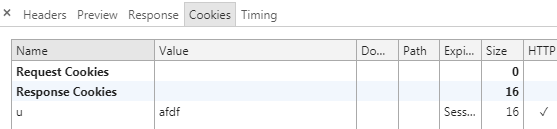
具体cookie：

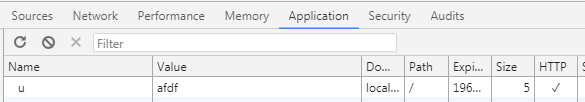
浏览器：设置》隐私设置和安全性。》内容设置》cookio>》浏览器是按域名存的

应用——购物车

在用户没有登录状态下选择商品，会将选择的商品信息放入cookie中，当客户端登录之后，会将请求发送给服务端，加到当前用户的信息里面。



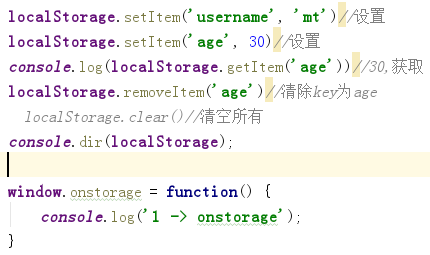




localStorage 的理解

在全局下有一个对象：localStorage

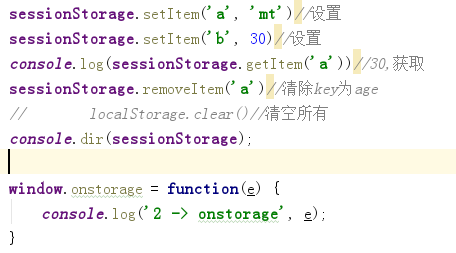
当我们在一个页面中设置localStorage的时候，其他同域下开启页面都会自动的更新localStorage，本质上是浏览器会在我们设置localStorage的时候通知其他的页面一个事件：storage事件



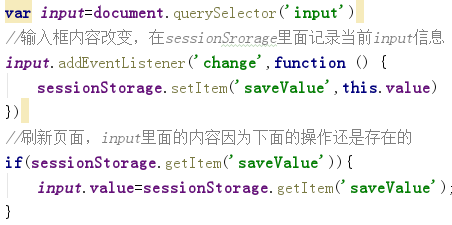
sessionStorage 的理解

在全局下有一个对象：sessionStorage

只能在当前设置的页面进行访问。当前页面关闭，就不存在了。



刷新页面，输入框内容还是存在：



应用：

购物车——和cookie一样，存储当前购物车信息，刷新页面，购物信息还在

音乐列表——

播放页和列表页。

点击列表页的歌曲，判断浏览器有没有打开播放页，没有打开，就打开播放页，将歌曲放到播放页的页面中播放。

再次点击列表页歌曲，判断浏览区有没有打开播放页，打开，就不再打开播放页，并将歌曲添加到播放页的播放列表中。

扫码登录的应用：

生成的码实质上是http链接，手机扫了链接地址，发送请求，请求登录地址。

手机扫，电脑马上登录。

电脑上生成扫码之后，一直在向服务器发ajax请求。手机扫码，服务器的是否登录状态改变，向电脑端发送改用户的内容。限定：手机有京东APP，手机京东已经登录，扫码之后才能同步到电脑。