

Cookie

王新阳

wxyyuppie@bjfu.edu.cn



Cookie

- **Cookie**的应用
- **Cookie**的基本思想
- **Cookie**的案例
- **Cookie**的基本方法





一、Cookie的应用

- 在网站访问过程中标识用户
 - 实现自动登录网站，避免存储用户名和密码
 - 实现用户访问网站日志的统计
- 记录用户个性化数据，实现个性化推荐
 - 实现站点定制，例如记录用户在某网站上访问的最后页面，在用户下次访问该网站时自动打开该页面
 - 实现广告按需投放，例如记录用户喜好的选择，推送符合其喜好的内容



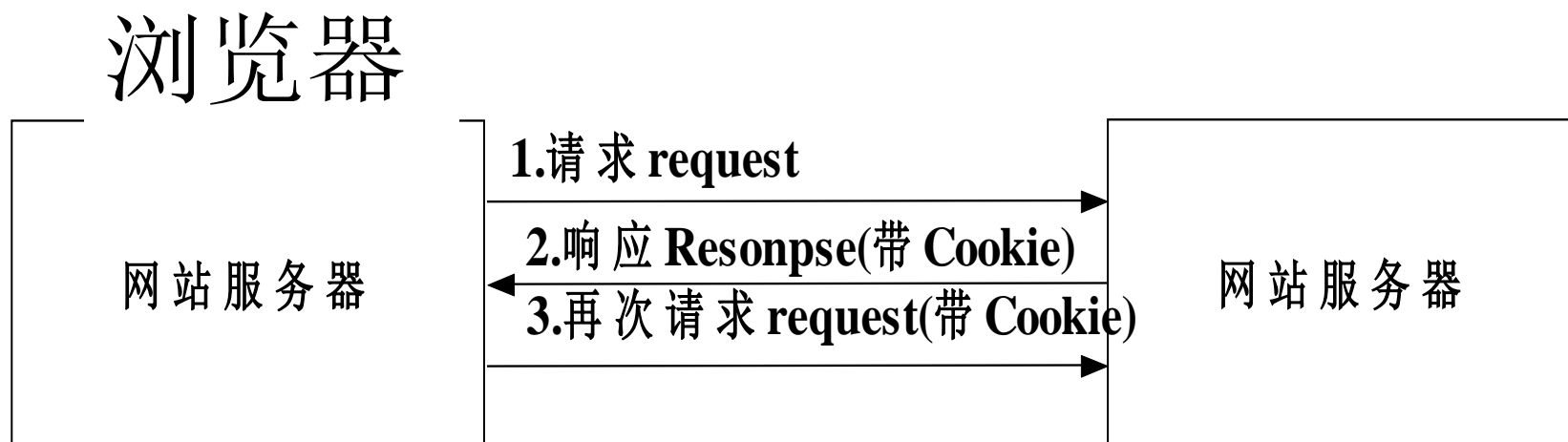
二、Cookie(小甜点)的基本思想

• Cookie的含义

- 是Web网站派送到浏览器端的“间谍”
 - ✓ 存储在客户端电脑上专门的文本文件中
- 是一个名称和值的数据对
 - ✓ 记录用户的个性化数据
 - 例如用户登录的用户名\密码，所作的选择等
 - 例如 用户访问的日期或URL等
- **Cookie**是一小块可以嵌入HTTP请求和响应的数据。它在服务器上产生，并作为响应头域的一部分返回给客户。浏览器在收到有**Cookie**的响应后，会把**Cookie**的内容以"关键字/值"的形式写入到一个专为存放**Cookie**的文本文件中。浏览器会把**Cookie**的信息与请求发送给服务器，这样服务器就可以再次读取**Cookie**中存放的数据了
- 在客户端浏览器再次访问该网站时，**Cookie**会自动随用户请求发回给网站
- 网站可以读取它放置的**Cookie**，从而读取到用户个性化数据，进行个性化处理



二、Cookie的基本思想



- 不同的浏览器存放的cookie位置不一样，不能通用



二、Cookie的基本思想

客户端Cookie的查看

- 查看某网站放置在客户端的**Cookie**
 - 在IE地址栏中输入 `javascript:alert(document.cookie)`
- 查看本机上的所有**Cookie**
 - Win7下----
 - 在C:\Users\用户名\AppData\Roaming\Microsoft\Windows\Cookies
 - 例如：C:\Users\user\AppData\Roaming\Microsoft\Windows\Cookies
 - IE的Cookie文件实际上就是一个txt文本文件，只不过换行符标记为Unix换行标记（0x0A），由于记事本对Unix换行标记不兼容，打开后内容全在一行看起来不方便，可以用EditPlus或UltraEdit-32打开
 - Chrome cookie存放路径：C:\Users\xxx\AppData\Local\Google\Chrome\User Data\Default，实际上是一个sqlite数据库文件，可以直接用sqlite数据库工具打开查看



二、Cookie的基本思想

- Cookie的格式说明
- Cookie 通常设置在 HTTP 头信息中（虽然 JavaScript 也可以直接在浏览器上设置一个 Cookie）。设置 Cookie 的 Servlet 会发送如下的头信息：
 - HTTP/1.1 200 OK
 - Date: Fri, 04 Feb 2000 21:03:38 GMT
 - Server: Apache/1.3.9 (UNIX) PHP/4.0b3
 - Set-Cookie: name=xyz; expires=Friday, 04-Feb-07 22:03:38 GMT;
 - path=/; domain=w3cschool.cc
 - Connection: close
 - Content-Type: text/html



二、Cookie的基本思想

<http://localhost:8080/ServletTest/CookieAgeTest>

localhost:8080/ServletTest/CookieAgeTest

Active Cookies

Cookie Name	Cookie Value
No cookies	

DevTools is now available in Chinese! [Always match Chrome's language](#) [Switch DevTools to Chinese](#) [Don't show again](#)

Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Performance insights

Filter ☐ Preserve log ☐ Disable cache No throttling ☐ Invert ☐ Hide data URLs All Fetch/XHR JS CSS img Media Font Doc WS Wasm Manifest Other ☐ Has blocked cookies ☐ Blocked Requests ☐ 3rd-party

5 ms 10 ms 15 ms 20 ms 25 ms 30 ms 35 ms 40 ms 45 ms 50 ms 55 ms 60 ms 65 ms 70 ms 75 ms 80 ms

Name X Headers Preview Response Initiator Timing Cookies

CookieAgeTest

Request URL: http://localhost:8080/ServletTest/CookieAgeTest
Request Method: GET
Status Code: 200
Remote Address: [::1]:8080
Referrer Policy: strict-origin-when-cross-origin

Response Headers

Content-Length: 324
Content-Type: text/html; charset=ISO-8859-1
Date: Sun, 18 Sep 2022 12:51:04 GMT
Set-Cookie: Session-Cookie-0=Cookie-Value-S0
Set-Cookie: Persistent-Cookie-0=Cookie-Value-P0; Max-Age=60; Expires=Sun, 18-Sep-2022
Set-Cookie: Session-Cookie-1=Cookie-Value-S1
Set-Cookie: Persistent-Cookie-1=Cookie-Value-P1; Max-Age=60; Expires=Sun, 18-Sep-2022
Set-Cookie: Session-Cookie-2=Cookie-Value-S2
Set-Cookie: Persistent-Cookie-2=Cookie-Value-P2; Max-Age=60; Expires=Sun, 18-Sep-2022

Request Headers

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Cache-Control: max-age=0
Connection: keep-alive
Host: localhost:8080
sec-ch-ua: "Google Chrome";v="105", "Not(A)Brand";v="8", "Chromium";v="105"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36

第一次没有Cookie

再次刷新出现Cookie

localhost:8080/ServletTest/CookieAgeTest

Active Cookies

Cookie Name	Cookie Value
Session-Cookie-0	Cookie-Value-S0
Session-Cookie-1	Cookie-Value-S1
Session-Cookie-2	Cookie-Value-S2

DevTools is now available in Chinese! [Always match Chrome's language](#) [Switch DevTools to Chinese](#) [Don't show again](#)

Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Performance insights

Filter ☐ Preserve log ☐ Disable cache No throttling ☐ Invert ☐ Hide data URLs All Fetch/XHR JS CSS img Media Font Doc WS Wasm Manifest Other ☐ Has blocked cookies ☐ Blocked Requests ☐ 3rd-party

5 ms 10 ms 15 ms 20 ms 25 ms 30 ms 35 ms 40 ms 45 ms 50 ms 55 ms 60 ms 65 ms 70 ms 75 ms

Name X Headers Preview Response Initiator Timing Cookies

CookieAgeTest

Request URL: http://localhost:8080/ServletTest/CookieAgeTest
Request Method: GET
Status Code: 200
Remote Address: [::1]:8080
Referrer Policy: strict-origin-when-cross-origin

Response Headers

Content-Length: 447
Content-Type: text/html; charset=ISO-8859-1
Date: Sun, 18 Sep 2022 12:54:00 GMT
Set-Cookie: Session-Cookie-0=Cookie-Value-S0
Set-Cookie: Persistent-Cookie-0=Cookie-Value-P0; Max-Age=60; Expires=Sun, 18-Sep-2022 12:55:00 GMT
Set-Cookie: Session-Cookie-1=Cookie-Value-S1
Set-Cookie: Persistent-Cookie-1=Cookie-Value-P1; Max-Age=60; Expires=Sun, 18-Sep-2022 12:55:00 GMT
Set-Cookie: Session-Cookie-2=Cookie-Value-S2
Set-Cookie: Persistent-Cookie-2=Cookie-Value-P2; Max-Age=60; Expires=Sun, 18-Sep-2022 12:55:00 GMT

Request Headers

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Cache-Control: max-age=0
Connection: keep-alive
Cookie: Session-Cookie-0=Cookie-Value-S0; Session-Cookie-1=Cookie-Value-S1; Session-Cookie-2=Cookie-Value-S2
Host: localhost:8080
sec-ch-ua: "Google Chrome";v="105", "Not(A)Brand";v="8", "Chromium";v="105"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36



二、Cookie的基本思想

- 如果浏览器被配置为存储 Cookie，它将会保留此信息直到有效日期。如果用户的浏览器指向任何匹配该 Cookie 的路径和域的页面，它会重新发送 Cookie 到服务器。
- GET / HTTP/1.0 Connection: Keep-Alive User-Agent: Mozilla/4.6 (X11; I; Linux 2.2.6-15apmac ppc) Host: zink.demon.co.uk:1126 Accept: image/gif, */* Accept-Encoding: gzip Accept-Language: en Accept-Charset: iso-8859-1,*,utf-8 Cookie: name=xyz



二、Cookie的基本思想

- 硬盘中的Cookies文件可以被[Web浏览器](#)读取，它的命令格式为：user@domain格式命名的，user是本地用户名，domain是所访问的网站的域名。
- 如Cookie:user@51talk.com/。要注意的是：硬盘中的Cookies属于文本文件，不是程序。



二、Cookie的基本思想

- Cookies的设置

IE的“工具/Internet选项” → “常规”选项卡 → “设置/查看文件”，查看所有保存到你电脑里的Cookies。



二、Cookie的基本思想

客户端Cookie的删除 (1)

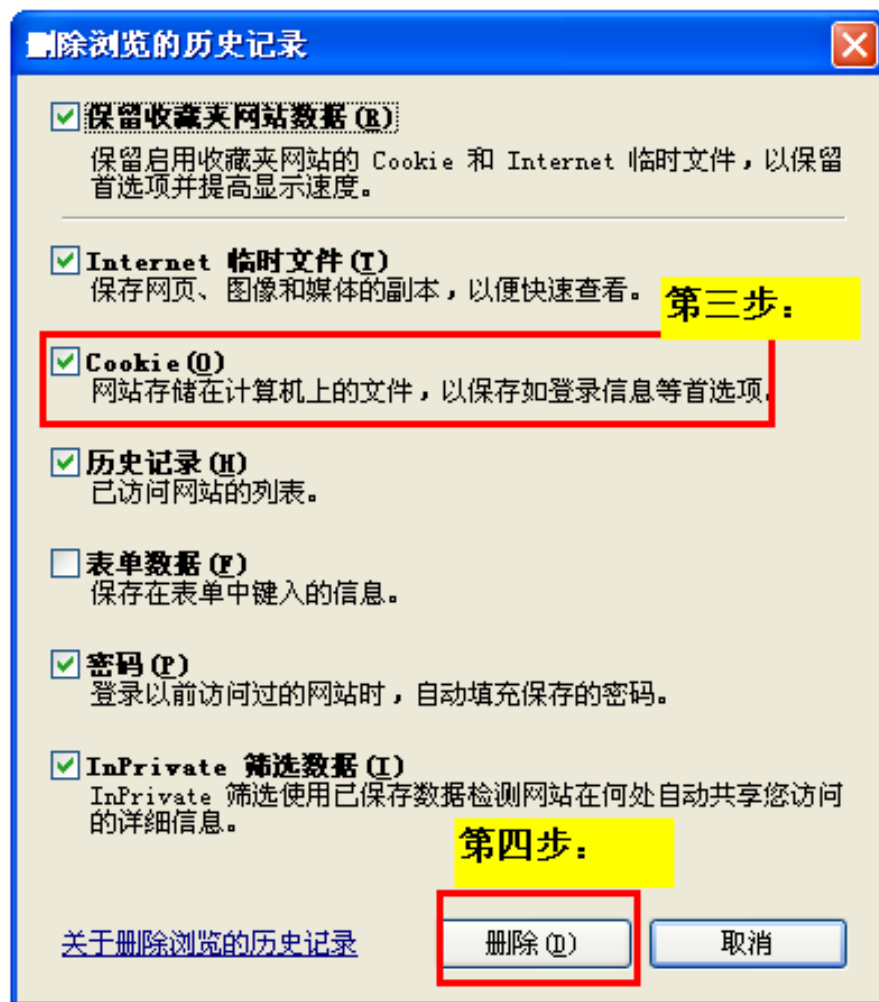
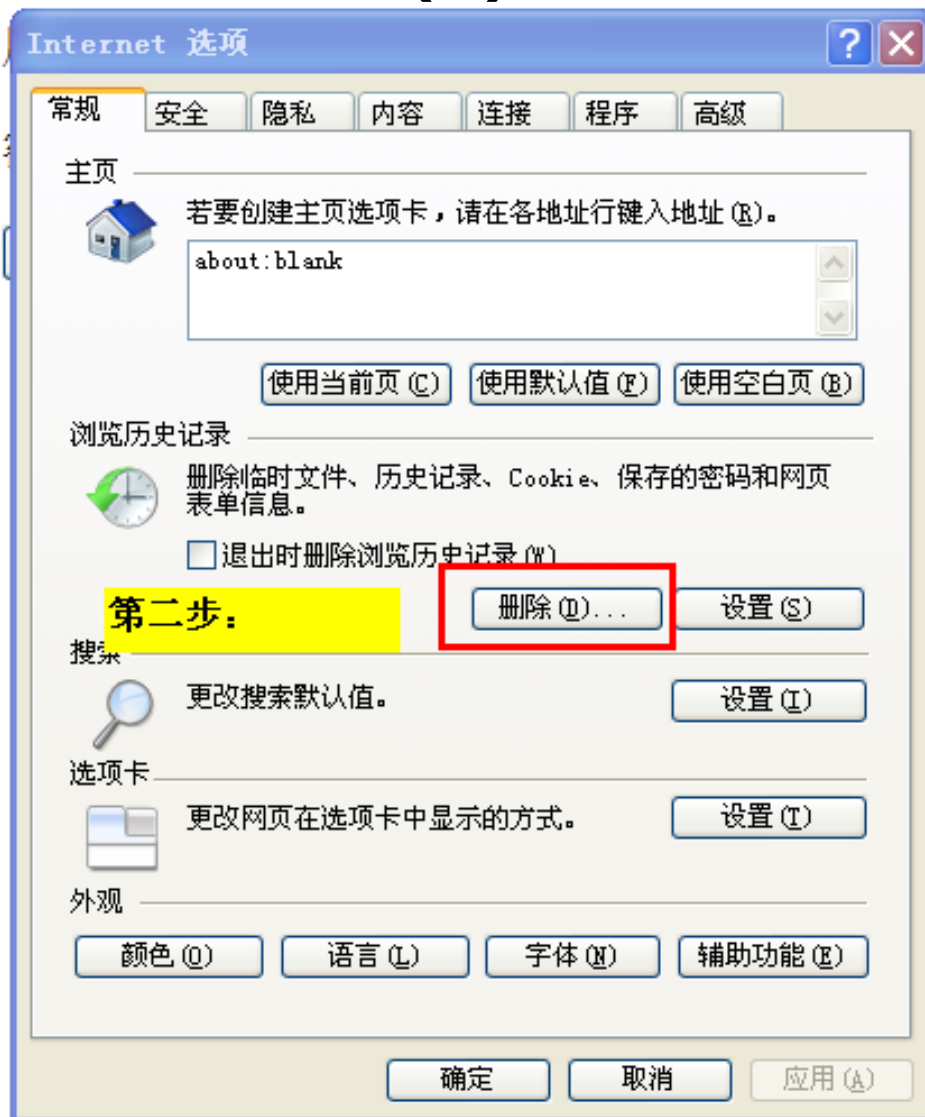
- IE浏览器→
工具→
Internet选项





二、Cookie的基本思想

客户端Cookie的删除 (1)

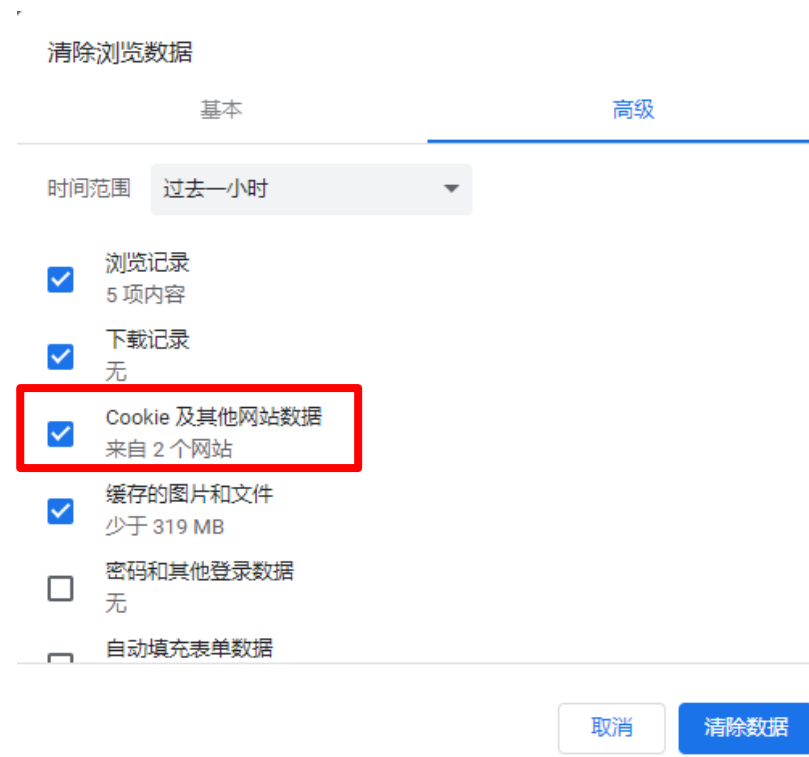
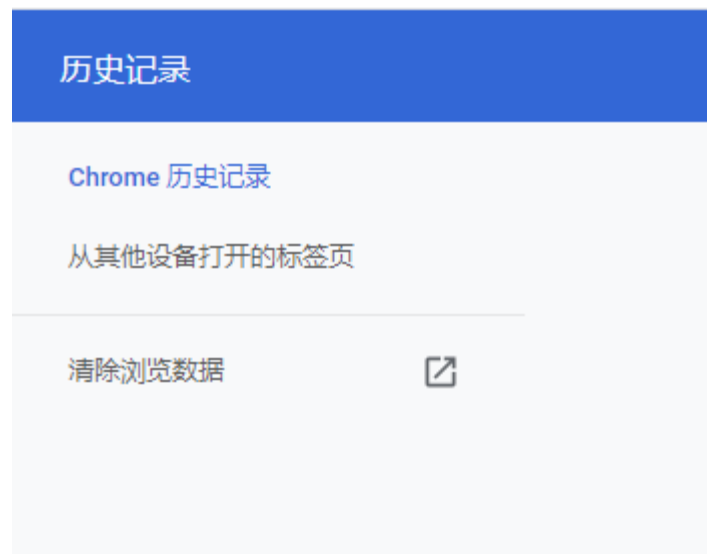




二、Cookie的基本思想

客户端Cookie的删除 (1)

- chrome浏览器→
历史记录→
清除浏览数据

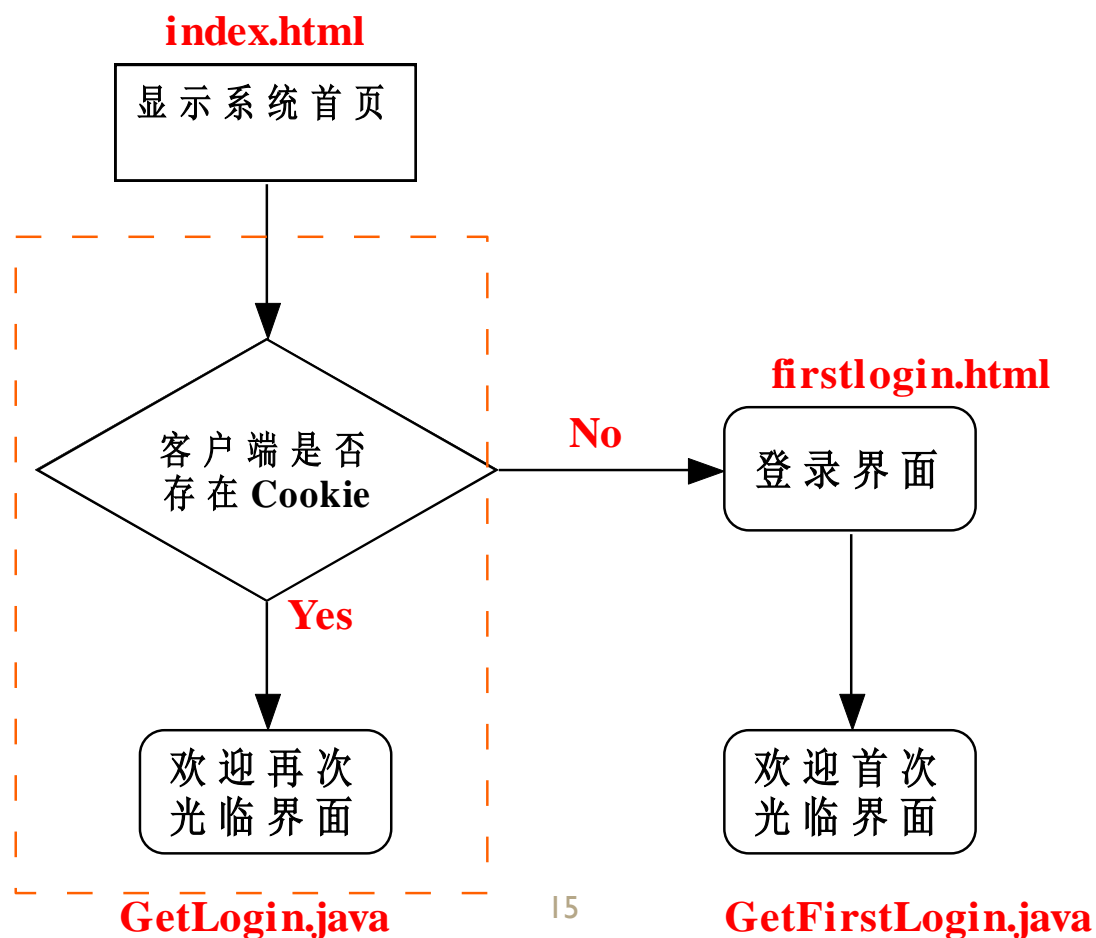




三、Cookie应用——例1实现用户自动登录

- 示例: **ServletTest.war**

- `cookie.GetLogin.java`, `cookie.GetFirstLogin.java`
- `index.html`, `firstlogin.html`





三、Cookie应用——例1实现用户自动登录

服务器从客户端读取Cookie内容——Getlogin.java

1. 先得到由浏览器所有Cookie对象组成的数组。
 - `Cookie[] cookies = request.getCookies();`
2. 再在这个数组中循环，调用每个对象的getName，直到找到想要的cookie为止。
 - `Cookie c = cookies[i];`
 - `if (c.getName().equals("c-username"))`
3. 根据应用程序的具体情况使用从Cookie中取出的这个值
 - `String username=c.getValue();`



三、Cookie应用——例1实现用户自动登录

服务器向客户端发送Cookie——GetFirstLogin.java

1. 创建**Cookie**对象，给出**cookie**的名称和**cookie**的值，二者都是字符串。
 - `Cookie usernameCookie = new Cookie("c-username", request.getParameter("username"));`
2. 将**Cookie**发送到客户端去
 - `response.addCookie(usernameCookie);`

事实上，

服务器创建的Cookie的值是“用户提供的值”，包括用户在表单中输入的值、用户访问的时间、URL地址等！



三、Cookie应用——例1实现用户自动登录

服务器向浏览器设置Cookie属性——一些常用的函数

1. 设置最大时效, 即告诉浏览器将**cookie**存储到磁盘上, 而非仅仅保存在内存中
 - //以秒为单位, 一个星期
 - `usernameCookie.setMaxAge(60*60*24*7)`
 - //删除某个Cookie
 - `C.setMaxAge(0);`
2. 修改**Cookie**的值
 - `c.setValue(string);`
 - `response.addCookie(c);`//需要把修改值发送回去
3. 设置可以提取**Cookie**的有效目录
 - 默认客户端的**Cookie**只对当前网页所在目录及其子目录中的网页有效
 - `c.setPath("/")`;//客户端的**Cookie**对整个Web服务器所有项目有效
 - `c.setPath("/ServletTest")`;//客户端的**Cookie**只对**ServletTest**项目有效



三、Cookie应用——例1实现用户自动登录

服务器从客户端读取Cookie属性——一些常用的函数

- 获取浏览器端所有的Cookie
 - `request.getCookies()`
- 获取一个Cookie的名字
 - `c.getName()`
- 获取一个Cookie的值
 - `c.getValue()`
- 获取一个Cookie的最大生存时长
 - `c.getMaxAge()`



三、Cookie应用——例2 观察Cookie的时效性

- 区分时效性
 - `c1.setMaxAge(-1)`,是默认值, 当关闭浏览器时就失效——会话Cookie
 - `c2.setMaxAge(300)`,意味着Cookie保存1分钟——持久Cookie
- 问题: 首次访问Servlet后,
 - 不关闭浏览器, 1分钟内, 再打开其它浏览器窗口, 有哪个Cookie?
 - 不关闭浏览器, 1分钟后, 再打开其它浏览器窗口, 有哪个Cookie?
 - 关闭浏览器, 1分钟后再次打开浏览器窗口, 有哪个Cookie?
 - 关闭浏览器, 1分钟内再次打开浏览器窗口, 有哪个Cookie?
- 示例: ServletTest.war
 - `cookie.CookieAgeTest.java`



三、Cookie应用——例2 观察Cookie的时效性

localhost:8080/ServletTest/CookieAgeTest

```
Cookie cookie = new Cookie("Session-Cookie-" + i, "Cookie-Value-S" + i);
response.addCookie(cookie);
cookie = new Cookie("Persistent-Cookie-" + i, "Cookie-Value-P" + i);

cookie.setMaxAge(60);
response.addCookie(cookie);
```

首次登录

再次登录

一分钟后再次登录

Active Cookies

Cookie Name	Cookie Value
No cookies	

Active Cookies

Cookie Name	Cookie Value
Session-Cookie-0	Cookie-Value-S0
Session-Cookie-1	Cookie-Value-S1
Session-Cookie-2	Cookie-Value-S2
Persistent-Cookie-0	Cookie-Value-P0
Persistent-Cookie-1	Cookie-Value-P1
Persistent-Cookie-2	Cookie-Value-P2

Active Cookies

Cookie Name	Cookie Value
Session-Cookie-0	Cookie-Value-S0
Session-Cookie-1	Cookie-Value-S1
Session-Cookie-2	Cookie-Value-S2

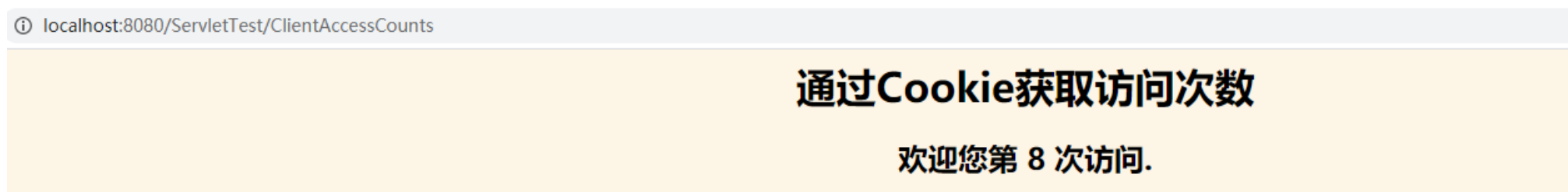


三、Cookie应用——例3 利用Cookie统计访问次数

- 更新cookie的值
 - 即把不同的值发送到相同的cookie中
 - 需要两步
 - ✓ setValue
 - ✓ response.addCookie
 - 不如直接：
 - ✓ 再新建一个同名的Cookie
 - ✓ response.addCookie即可
- 示例 **ServletTest.jar**
 - cookietest.LongLivedCookie.java
 - cookietest.ClientAccessCounts.java



三、Cookie应用——例3 利用Cookie统计访问次数





小结

- **Cookie的本质**

- 将名称/值对从服务器发送到浏览器，并在之后访问相同的页面、站点或域时返回给服务器。

- **Cookie的应用**

- 保存用户提供或使用的私有数据，方便他用
- 也存在不安全因素

- **设置cookie**

- 调用**Cookie**的构造函数，设置时效
- 调用**response.addCookie**

- **读取cookie**

- 调用**request.getCookie**，检查得到的结果是否为**null**，
- 在数组中查找匹配的名称，使用相关的值



思考和练习

- 如何实现用户在第一次以后访问某网站时，首先可以看到自己上次登录的时间和离开的时间？
- 如何实现用户输入简历后，每次再访问注册页面时自动跳转到已登录页面（该页面显示简历信息）？



感谢聆听

Thanks For Your Listening!