



会话技术 Cookie&Session

教学导航

教学目标	案例一、记录用户的上次访问时间---cookie 案例二、实现验证码的校验----session
教学方法	

一、会话技术简介

1. 存储客户端的状态

由一个问题引出今天的内容，例如网站的购物系统，用户将购买的商品信息存储到哪里？因为 Http 协议是无状态的，也就是说每个客户访问服务器端资源时，服务器并不知道该客户端是谁，所以需要会话技术识别客户端的状态。会话技术是帮助服务器记住客户端状态（区分客户端）

2. 会话技术

从打开一个浏览器访问某个站点，到关闭这个浏览器的整个过程，成为一次会话。会话技术就是记录这次会话中客户端的状态与数据的。

会话技术分为 Cookie 和 Session：

Cookie：数据存储客户端本地，减少服务器端的存储的压力，安全性不好，客户端可以清除 cookie

Session：将数据存储到服务器端，安全性相对好，增加服务器的压力

二、Cookie 技术

Cookie 技术是将用户的数据存储到客户端的技术，我们分为两方面学习：

第一，服务器端怎样将一个 Cookie 发送到客户端

第二，服务器端怎样接受客户端携带的 Cookie

1. 服务器端向客户端发送一个 Cookie

1) 创建 Cookie :

Cookie cookie = new Cookie(String cookieName,String cookieValue);

示例 :

```
Cookie cookie = new Cookie("username" , "zhangsan");
```

那么该 cookie 会以响应头的形式发送给客户端 :

```
Set-Cookie: "name=zhangsan"
```

注意 : Cookie 中不能存储中文

2) 设置 Cookie 在客户端的持久化时间 :

cookie.setMaxAge(int seconds); ---时间秒

注意 : 如果不设置持久化时间 , cookie 会存储在浏览器的内存中 , 浏览器关闭 cookie 信息销毁 (会话级别的 cookie) , 如果设置持久化时间 , cookie 信息会被持久化到浏览器的磁盘文件里

示例 :

```
cookie.setMaxAge(10*60);
```

设置 cookie 信息在浏览器的磁盘文件中存储的时间是 10 分钟 , 过期浏览器自动删除该 cookie 信息

3) 设置 Cookie 的携带路径 :

cookie.setPath(String path);

注意 : 如果不设置携带路径 , 那么该 cookie 信息会在访问产生该 cookie 的 web 资源所在的路径都携带 cookie 信息

示例 :

```
cookie.setPath("/WEB16");
```

代表访问 WEB16 应用中的任何资源都携带 cookie

```
cookie.setPath("/WEB16/cookieServlet");
```

代表访问 WEB16 中的 cookieServlet 时才携带 cookie 信息

4) 向客户端发送 cookie :

response.addCookie(Cookie cookie);

5) 删除客户端的 cookie :

如果想删除客户端的已经存储的 cookie 信息 ,那么就使用**同名同路径的持久化时间**为 0 的 cookie 进行覆盖即可

2. 服务器端怎么接受客户端携带的 Cookie

cookie 信息是以请求头的方式发送到服务器端的：

```
Cookie: "name=zhangsan"
```

1) 通过 request 获得所有的 Cookie：

```
Cookie[] cookies = request.getCookies();
```

2) 遍历 Cookie 数组，通过 Cookie 的名称获得我们想要的 Cookie

```
for(Cookie cookie : cookies){  
    if(cookie.getName().equals(cookieName)){  
        String cookieValue = cookie.getValue();  
    }  
}
```

三、Session 技术

Session 技术是将数据存储在服务端的技术，会为每个客户端都创建一块内存空间存储客户的数据，但客户端需要每次都携带一个标识 ID 去服务器中寻找属于自己的内存空间。所以说 Session 的实现是基于 Cookie，Session 需要借助于 Cookie 存储客户的唯一性标识 JSESSIONID

在 Session 这我们需要学习如下三个问题：

怎样获得属于本客户端的 session 对象（内存区域）？

怎样向 session 中存取数据（session 也是一个域对象）？

session 对象的生命周期？

1. 获得 Session 对象

```
HttpSession session = request.getSession();
```

此方法会获得专属于当前会话的 Session 对象，如果服务器端没有该会话的 Session

对象会创建一个新的 Session 返回，如果已经有了属于该会话的 Session 直接将已有的 Session 返回（实质就是根据 JSESSIONID 判断该客户端是否在服务器上已经存在 session 了）

2. 怎样向 session 中存取数据（session 也是一个域对象）

Session 也是存储数据的区域对象，所以 session 对象也具有如下三个方法：

```
session.setAttribute(String name,Object obj);  
session.getAttribute(String name);  
session.removeAttribute(String name);
```

3. Session 对象的生命周期（面试题/笔试题）

创建：第一次执行 request.getSession()时创建

销毁：

- 1) 服务器（非正常）关闭时
- 2) session 过期/失效（默认 30 分钟）

问题：时间的起算点 从何时开始计算 30 分钟？

从不操作服务器端的资源开始计时

可以在工程的 web.xml 中进行配置

```
<session-config>  
    <session-timeout> 30</session-timeout>  
</session-config>
```

- 3) 手动销毁 session

```
session.invalidate();
```

作用范围：

默认在一次会话中，也就是说在，一次会话中任何资源公用一个 session 对象

面试题：浏览器关闭，session 就销毁了？ 不对

总结：

重点：图

会话技术：

Cookie 技术：存到客户端

发送 cookie

```
Cookie cookie = new Cookie(name,value)
cookie.setMaxAge(秒)
cookie.setPath()
response.addCookie(cookie)
```

获得 cookie

```
Cookie[] cookies = request.getCookies();
cookie.getName();
cookie.getValue();
```

Session 技术：存到服务器端 借助 cookie 存储 JSESSIONID

```
HttpSession session = request.getSession();
```

```
setAttribute(name,value);
```

```
getAttribute(name);
```

session 生命周期

创建：第一次指定 request.getSession();

销毁：服务器关闭、session 失效/过期、手动 session.invalidate();

session 作用范围：默认一会话中



黑马程序员
www.itheima.com

传智播客旗下
高端IT教育品牌

改变中国IT教育，我们正在行动

传智播客Java学院