

关于JSP中基于Session的在线用户统计分析计算机等级考试  
PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/646/2021\\_2022\\_\\_E5\\_85\\_B3\\_E4\\_BA\\_8EJSP\\_E4\\_c97\\_646044.htm](https://www.100test.com/kao_ti2020/646/2021_2022__E5_85_B3_E4_BA_8EJSP_E4_c97_646044.htm) JSP作为后起之秀能够在服务器编程环境中占据一定地位，是和它良好支持一系列业界标准密切相关的。Session就是它提供的基础设施之一。作为一个程序员，你可以不介意具体在客户端是如何实现，就方便的实现简单的基于session的用户管理。现在对于处理在线用户，有几种不同的处理方法。一种是页面刷新由用户控制，服务器端控制一个超时时间比如30分钟，到了时间之后用户没有动作就被踢出。这种方法的优点是，如果用户忘了退出，可以防止别人恶意操作。缺点是，如果你在做一个很耗时间的东西，超过了这个时间限制，submit的时候可能要再次面临登陆。如果原来的页面又是强制失效的话，就有可能丢失你做的东西。在实现的角度来看，这是最简单的，Server端默认实现的就是这样的模式。另一种方式是，站点采用框架结构，有一个Frame或者隐藏的iframe在不断刷新，这样你永远不会被踢出，但是服务器端为了判断你是否在线，需要定一个发呆时间，如果超过这个发呆时间你除了这个自动刷新的页面外没有刷新其他页面的话，就认为你已经不在线了。采取这种方式的典型是xici.net。他的优点是可以通过不断的刷新实现一些类似server-push的功能，比如网友之间发送消息。不管哪一种模式，为了实现浏览当前所有的在线用户，还需要做一些额外的工作。Servlet API中没有得到Session列表的API。可以利用的是Listener。Servlet 2.2和2.3规范在这里略微有一些不一样。2.2中HttpSessionBindingListener可以实现当

一个HTTPSession中的Attribute变化的时候通知你的类。而2.3中还引入了HttpSessionAttributeListener.鉴于我使用的环境是Visual age for Java 4和JRun server 3.1,他们还不直接支持Servlet 2.3的编程，这里我用的是HttpSessionBindingListener. 需要做的事情包括做一个新的类来实现HttpSessionBindingListener接口。这个接口有两个方法：

```
public void valueBound(HttpSessionBindingEvent event) public void valueUnbound(HttpSessionBindingEvent event)
```

当你执行Session.addAttribute(String,Object)的时候，如果你已经把实现HttpSessionBindingListener接口的类加入为Attribute,Session会通知你的类，调用你的valueBound方法。相反，Session.removeAttribute方法对应的是valueUnbound方法。

```
public class HttpSessionBinding implements javax.servlet.http.HttpSessionBindingListener { ServletContext application = null. public HttpSessionBinding(ServletContext application) { super(). if (application ==null) throw new IllegalArgumentException("Null application is not accept."). this.application = application. } public void valueBound(javax.servlet.http.HttpSessionBindingEvent e) { Vector activeSessions = (Vector) application.getAttribute("activeSessions"). if (activeSessions == null) { activeSessions = new Vector(). } JDBCUser sessionUser = (JDBCUser)e.getSession().getAttribute("user"). if (sessionUser != null) { activeSessions.add(e.getSession()). } application.setAttribute("activeSessions",activeSessions). } public void valueUnbound(javax.servlet.http.HttpSessionBindingEvent e) {
```

```
JDBCUser sessionUser =  
(JDBCUser)e.getSession().getAttribute("user"). if (sessionUser ==  
null) { Vector activeSessions = (Vector)  
application.getAttribute("activeSessions"). if (activeSessions != null)  
{ activeSessions.remove(e.getSession().getId()).  
application.setAttribute("activeSessions",activeSessions). } } }
```

100Test 下载频道开通，各类考试题目直接下载。详细请访问  
[www.100test.com](http://www.100test.com)