<script language="javascript" type="">

function countDown(secs){

 tiao.innerText=secs;

 if(--secs>0)

  setTimeout("countDown("+secs+")",1000);

 }

 countDown(3);

</script>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

  <head>

      <META HTTP-EQUIV="Refresh" CONTENT="秒数; URL=转向页面"/>

  </head>

  <body>

。。。。

  </body>

</html>

<%response.serHeader("Refresh","3,url=要跳转到的页面地址");%>

**JSP页面跳转的几个方法**

2010-11-06  [mickyliang](http://www.360doc.com/userhome/2898817" \t "_blank)  转自 [冷刃寒冰](http://www.360doc.com/userhome/654841)

[转藏到我的图书馆](javascript:void(0);)

[微信](javascript:void(0);)分享：

|  |
| --- |
| **1. RequestDispatcher.forward()** 在服务器端起作用,当使用forward()时,Servlet engine传递HTTP请求从当前的Servlet或者是JSP到另外的一个Servlet、JSP 或普通HTML文件,也即你的form提交至a.jsp,在a.jsp用到了forward()重定向至b.jsp,此时form提交的所有信息在 b.jsp都可以获得,参数自动传递. 但forward()无法重定向至有frame的jsp文件,可以重定向至有frame的html文件,同时forward()无法在后面带参数传递,比 如servlet?name=frank,这样不行,可以程序内通过response.setAttribute("name",name)来传至下一个 页面。 　**重定向后浏览器地址栏URL不变。**  　　例：在servlet中进行重定向 　　public void doPost(HttpServletRequest request,HttpServletResponse response) 　　                              throws ServletException,IOException{ 　　response.setContentType("text/html; charset=gb2312"); 　　ServletContext sc = getServletContext(); 　　RequestDispatcher rd = null; 　　rd = sc.getRequestDispatcher("/index.jsp"); //定向的页面 　　rd.forward(request, response); 　　} 　　通常在servlet中使用，不在jsp中使用。  **2. response.sendRedirect()**  　　在用户的浏览器端工作,sendRedirect()可以带参数传递,比如servlet?name=frank传至下个页面,同时它可以重定向至不同的主机上,sendRedirect()可以重定向有frame.的jsp文件. 　　重定向后在浏览器地址栏上会出现重定向页面的URL。 　　例：在servlet中重定向 　　public void doPost(HttpServletRequest request,HttpServletResponse response) 　　                              throws ServletException,IOException{ 　　response.setContentType("text/html; charset=gb2312"); 　　response.sendRedirect("/index.jsp"); 　　} 　　**由于response是jsp页面的隐含对象，故在jsp页面中可用response.sendRedirect()直接实现重定位。**  　　注意： 　　(1) 使用response.sendRedirect时，前面不能有HTML输出； 这并不是绝对的，不能有HTML输出其实是指不能有HTML被送到了浏览器。事实上现在的server都有cache机制，一般在8K（我是说 JSP　SERVER），这就意味着，除非你关闭了cache，或者你使用了out.flush()强制刷新，那么在使用sendRedirect之前， 有少量的HTML输出也是允许的。 　　(2) response.sendRedirect之后，应该紧跟一句return。 　　我们已经知道response.sendRedirect是通过浏览器来做转向的，所以只有在页面处理完成后，才会有实际的动作。既然你已经要做转向了，那么后的输出还有什么意义呢？而且有可能会因为后面的输出导致转向失败。 　　比较： 　　(1) Dispatcher.forward()是容器中控制权的转向，在客户端浏览器地址栏中不会显示出转向后的地址； 　　(2) response.sendRedirect()则是完全的跳转，浏览器将会得到跳转的地址，并重新发送请求链接。这样，从浏览器的地址栏中可以看到跳转后的链接地址。 　　前者更加高效，在前者可以满足需要时，尽量使用RequestDispatcher.forward()方法。 　　在有些情况下，比如，需要跳转到一个其它服务器上的资源，则必须使HttpServletResponse.sendRequest()方法。  **3. ＜jsp:forward page="" /＞**  　　它的底层部分是由RequestDispatcher来实现的，因此它带有RequestDispatcher.forward()方法的印记。 　　如果在之前有很多输出,前面的输出已使缓冲区满,将自动输出到客户端,那么该语句将不起作用,这一点应该特别注意。 　　注意：它不能改变浏览器地址，刷新的话会导致重复提交  **4. 修改HTTP header的Location属性来重定向**  　　通过设置直接修改地址栏来实现页面的重定向。 　　jsp文件代码如下： 　　＜% 　　response.setStatus(HttpServletResponse.SC\_MOVED\_PERMANENTLY); 　　String newLocn = "/newpath/jsa.jsp"; 　　response.setHeader("Location",newLocn); 　　%＞  **5. JSP中实现在某页面停留若干秒后,自动重定向到另一页面**  　　在html文件中，下面的代码： 　　＜meta http-equiv="refresh" content="300; url=target.jsp"＞ 　　它的含义：在5分钟之后正在浏览的页面将会自动变为target.html这一页。代码中300为刷新的延迟时间，以秒为单位。targer.html为你想转向的目标页,若为本页则为自动刷新本页。 　　由上可知，可以通过setHeader来实现某页面停留若干秒后,自动重定向到另一页面。代码： 　　String content=stayTime+";URL="+URL; 　　response.setHeader("REFRESH",content); |